Application Note

Methods for Mitigating ADC Memory Cross-Talk

Devin Cottier

ABSTRACT

High-speed multiplexed analog-to-digital converters (ADCs) like those found on C2000™ series microcontrollers enable fast sensing of multiple feedback signals in real-time control applications. The signal conditioning circuits for these feedback signals should be carefully designed and evaluated to ensure that adequate settling is achieved in the time allocated for the ADC's sample-and-hold (S+H). In cases where adequate settling is not achieved in the hardware design of the signal conditioning circuits, memory cross-talk issues can arise in the system. This application report reviews the causes and symptoms of memory cross-talk and then presents two possible strategies for mitigating the memory cross-talk error: dedicating an ADC to the affected signal (resulting in only self cross-talk) and sampling ground before the affected signal (transforming the memory cross-talk into a gain error, which can subsequently be calibrated out of the system).

Project collateral discussed in this document can be downloaded from the following URL: https://www.ti.com/lit/zip/spracw9.

Table of Contents

1 Introduction ... 3
1.1 Memory Cross-Talk Challenges ... 3
1.2 Resources for Signal Conditioning Circuit Design ... 3
2 Review of ADC Input Settling ... 4
2.1 Mechanism of ADC Input Settling ... 5
2.2 Symptoms of Inadequate Settling ... 5
2.3 C2000 ADC Architecture ... 7
3 Problem Statement ... 8
3.1 Example System ... 8
3.2 S+H Settling Analysis ... 10
3.3 Charge-Sharing Analysis ... 11
3.4 Problem Summary ... 12
4 Dedicated ADC Sampling ... 13
4.1 Dedicated ADC Concept ... 13
4.2 Settling Mechanism for Dedicated ADC ... 14
4.3 Design Flow for Dedicated ADC ... 15
4.4 Simulating Settling Performance for a Dedicated ADC Circuit ... 16
5 Pre-Sampling VREFLO ... 19
5.1 VREFLO Sampling Concept ... 19
5.2 Properties of VREFLO Sampling Method Error ... 20
5.3 Gain Error Compensation ... 21
5.4 VREFLO Sampling Design Flow ... 21
5.5 Discussion of VREFLO Sampling Sequences ... 22
6 Summary ... 23
7 References ... 23
8 Revision History ... 23

List of Figures

Figure 2-1. Settling of the ADC S+H Capacitor ... 5
Figure 2-2. Sequence of Samples With C_H Retained Between Conversions ... 6
Figure 2-3. Sequence of Samples With C_H Discharged Between Conversions ... 6
Figure 2-4. Sequence of Multiplexed Samples ... 7

Figure 3-1. Example System Timings....8
Figure 3-2. Example System Schematic for V2....9
Figure 3-3. Settling Simulation for V2 Circuit With 100 ns S+H....11
Figure 3-4. Example System Memory Cross-talk....12
Figure 4-1. Timings for Dedicated ADC Method....13
Figure 4-2. Simulation Schematic for V2 on Dedicated ADC....14
Figure 4-3. Settling Simulation for V2 Circuit on Dedicated ADC....15
Figure 4-4. Schematic for AC Simulation for Dedicated ADC....16
Figure 4-5. Settling Simulation for 60Hz AC Input....17
Figure 4-6. Settling Simulation for 6kHz AC Input....18
Figure 5-1. Timings for VREFLO Sampling Method....19
Figure 5-2. Schematic for VREFLO Sampling Simulation....20
Figure 5-3. Timings for VREFLO Sampling for All Channels....22

List of Tables

Table 3-1. Settling Time for V2 Circuit....10
Table 5-1. Sampled Voltage vs. Vin With VREFLO Cross-Talk....20

Trademarks

C2000™ and TINA-TI™ are trademarks of Texas Instruments.
PSPICE® is a registered trademark of Cadence Design Systems, Inc.
All trademarks are the property of their respective owners.

1 Introduction

Proper design and evaluation of the signal conditioning circuits driving an ADC input is an essential step for ensuring good ADC performance. Of primary concern is the settling performance: can the circuit driving the ADC input charge the ADC's sample-and-hold (S+H) circuit to within acceptable tolerance of the voltage to be sensed in the allocated S+H time? If not, the sensed voltage will have some error. Furthermore, in systems where the ADC is scanning through multiple multiplexed inputs to be sampled (a common use case in real-time control applications), this error will show up as memory cross-talk: the value of the previous conversion in the sampling sequence will effect the results of the signal that was sampled with inadequate S+H settling.

1.1 Memory Cross-Talk Challenges

Due to the nature of the memory cross-talk, the error can appear systematic (if the previous conversion result is correlated to the conversion result experiencing the error) or random (if the previous conversion result is changing asynchronously with respect to the signal experiencing the cross-talk). The error can also appear constant if and when the previously converted channel is not changing very fast.

Because the error can manifest in many different ways, it can be very hard to diagnose in a system. Furthermore, it is difficult to compensate for memory cross talk: oversampling and averaging is not very effective in reducing the magnitude since the error is not fully random, but the error is also usually not systematic enough to be calibrated out of the system as if it were a fixed gain or offset error.

1.2 Resources for Signal Conditioning Circuit Design

Since memory cross-talk error is hard to identify and hard to resolve, care should be taken to ensure good settling performance when designing ADC driving circuits. In addition, evaluating the input settling should be one of the first checks when an unknown ADC performance issue is observed in a real-time control system.

Thankfully, a number of good resources exist to help with designing and evaluating ADC driving circuits for S+H settling performance for C2000 MCUs:

1.2.1 TI Precision Labs - SAR ADC Input Driver Design Series

TI precision labs has provided an excellent seven-part video series that demonstrates how to design the input drivers for a SAR ADC.

Link to Video Training Series: TI Precision Labs - SAR ADC Input Driver Design

1.2.2 Analog Engineer's Calculator

The analog engineer's calculator tool provides a variety of very useful GUI-based calculation tabs to assist with common analog circuit design tasks. The TI Precision labs methodology takes advantage of the Data Converters → ADC SAR Drive calculator.

Link to Tool Folder: Analog Engineer's Calculator

1.2.3 Related Application Reports

TI provides several application reports that pertain to the design and evaluation of ADC input circuits on C2000 real-time MCU devices. These application reports are listed along with brief descriptions.

Application Report Title	Schematic Capture and Simulation Tool	Purpose
ADC Input Circuit Evaluation for C2000 MCUs (TINA-TI)	TINA-TI	Describes how to design and evaluate a traditional high-speed ADC driving circuit. Input circuit includes an op-amp. Appropriate when sampling a high bandwidth signal near the maximum sample rate of the ADC.
ADC Input Circuit Evaluation for C2000 MCUs (PSpice for TI)	PSpice for TI	
Charge-Sharing Driving Circuits for C2000 ADCs (TINA-TI)	TINA-TI	Describes how to design and evaluate a charge-sharing ADC driving circuit. Input circuit may or may not include an op-amp. Appropriate when sampling a signal with a sufficiently low bandwidth and sample rate.
Charge-Sharing Driving Circuits for C2000 ADCs (PSpice for TI)	PSpice for TI	
Methods for Mitigating ADC Memory Cross-Talk	TINA-TI	Describes strategies to mitigate memory cross-talk error when the input circuit design does not achieve adequate settling. Applicable under special circumstances.

1.2.4 TINA-TI SPICE-Based Analog Simulation Program

TI provides the TINA-TI™ application to TI customers to allow easy SPICE-based simulation of circuits utilizing TI products. The TI Precision Labs input settling design methodology uses TINA-TI to perform a number of simulations to refine and verify the input circuit design for proper settling.

Link to Tool Folder: TINA-TI

1.2.5 PSPICE for TI

TI also provides the PSPICE® for TI tool to enable SPICE-based simulation of circuits utilizing TI products. This can be used as an alternative to TINA-TI for input settling simulations, if desired.

Link to Tool Folder: PSpice for TI

1.2.6 ADC Input Circuit Evaluation for C2000 MCUs

ADC Input Circuit Evaluation for C2000 MCUs adapts the TI Precision Labs SAR ADC Input Driver video series specifically for C2000 series MCUs. In addition to C2000-specific input circuit design and evaluation methodology, pre-made TINA-TI models for a number of C2000 devices are included.

1.2.7 Charge-Sharing Driving Circuits for C2000 ADCs

Charge-Sharing Driving Circuits for C2000 ADCs presents an alternative ADC input driving circuit design methodology. The charge sharing methodology uses a much larger input capacitor to eliminate the need for a high speed driving op-amp in the input design. The trade-off for this simplification is that charge sharing designs support lower ADC sample rates and lower signal bandwidth.

2 Review of ADC Input Settling

The following sections review the need to evaluate input settling in ADCs and the errors that can occur when proper input settling is not achieved.

2.1 Mechanism of ADC Input Settling

To convert a sensed analog voltage to a digital conversion result, the ADC first must accurately capture the applied input voltage into its sample-and-hold circuit (S+H). As shown in Figure 2-1, this entails charging the internal ADC S+H capacitor (C_h) to within some acceptable tolerance (typically 0.5 LSBs) of the applied voltage within the configured acquisition window time (also referred to as the S+H time).

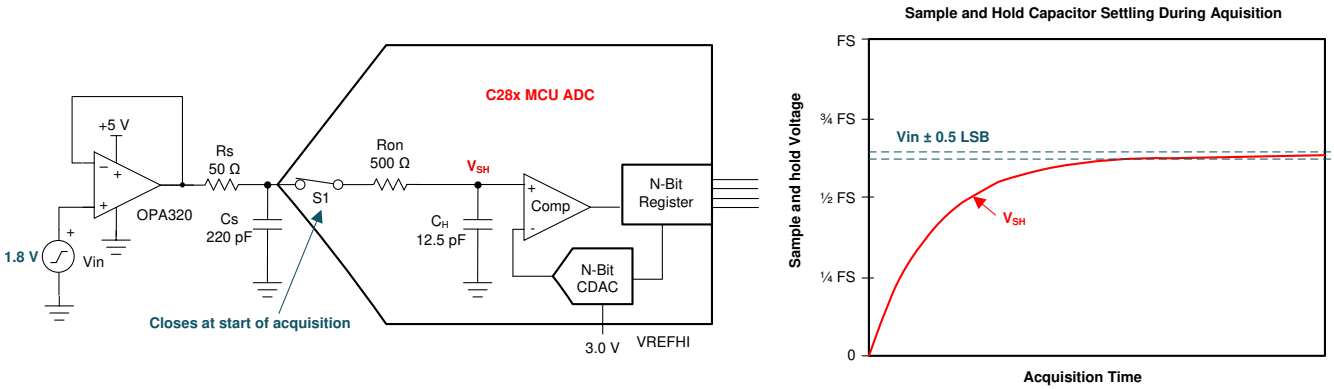

Figure 2-1. Settling of the ADC S+H Capacitor

Quickly charging C_h to the applied voltage is complicated by the finite bandwidth and settling time of the external ADC driver circuit and of the settling time of the internal ADC S+H circuit. In Figure 2-1, the driver is show as an op-amp (OPA320), which has a finite bandwidth, and the driver circuit also has intentionally placed source resistance (R_s) and intentionally placed source capacitance (Cs) which have a finite settling time determined by their RC time constant. Note that other circuit topologies are possible for driving the ADC, and these circuits may have additional components that need to be modeled to ensure appropriate settling time. These components could include unintentional parasitics such as the output impedance of a sensor or the effective source resistance of a voltage divider. Figure 2-1 also shows that the ADC has an internal parasitic switch resistance (R_{on}). This, along with C_h, provides an additional RC time constant that limits settling speed.

2.2 Symptoms of Inadequate Settling

Once a voltage has been captured into the S+H capacitor, the ADC will translate this voltage into a digital conversion result during the conversion phase. The CPU can then use this result to control or monitor the system. However, if the captured voltage does not accurately represent the applied voltage due to settling error, the final conversion result will have errors even if the ADC conversion process is perfect.

These settling errors will manifest differently depending on whether the ADC is sampling the same channel repeatedly or scanning through multiple channels in a sequence. The settling errors will also manifest differently depending on the starting voltage on the S+H capacitor at the beginning of the acquisition phase. Some ADC architectural implementations will have a starting S+H voltage close to the previously sampled voltage while other architectures will usually start the acquisition phase with a discharged S+H capacitor.

2.2.1 Distortion

In the case where the ADC is repeatedly sampling the same signal, settling error typically manifests as distortion of the input signal. In architectures where the S+H voltage starts near the previously sampled voltage, slow moving portions of the input signal will settle better than fast moving portions. An architecture where sequential samples begin their settling from the voltage sampled and held in the previous conversion is illustrated in Figure 2-2.

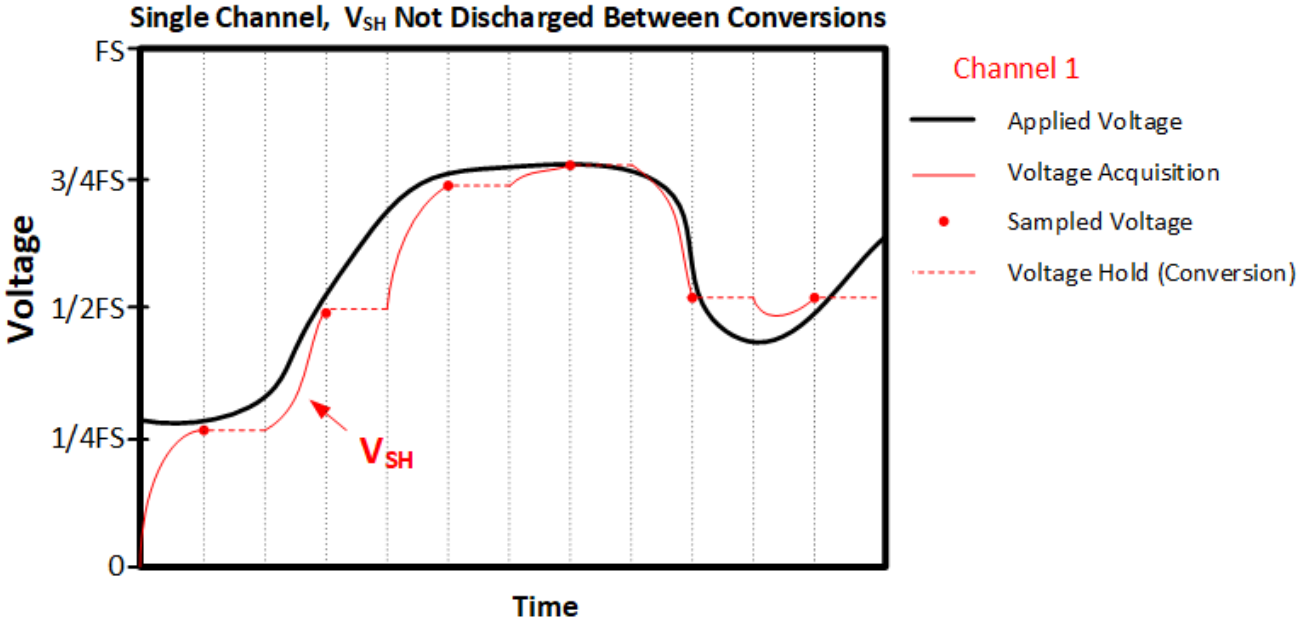

Figure 2-2. Sequence of Samples With C_H Retained Between Conversions

In architectures where the S+H capacitor starts each acquisition phase discharged, higher input voltages will have worse settling, resulting in distorted scaling of the signal. An architecture where sequential samples always beginning their settling from near zero-scale is illustrated in Figure 2-3.

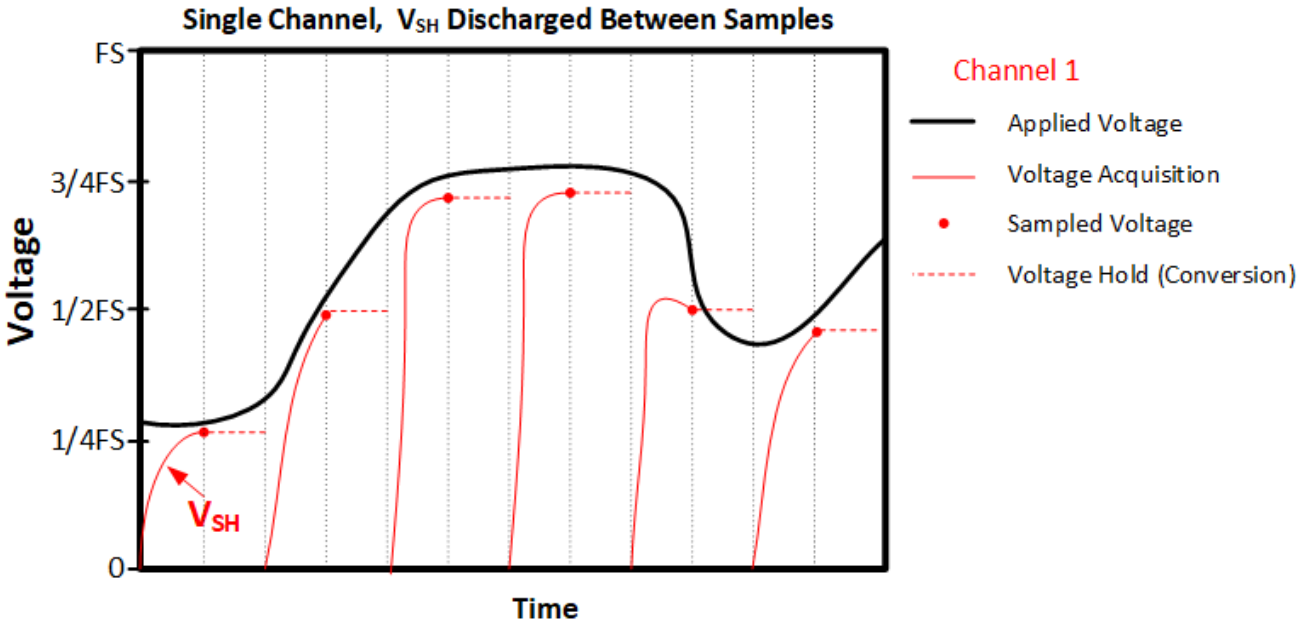

Figure 2-3. Sequence of Samples With C_H Discharged Between Conversions

2.2.2 Memory Cross-Talk

In many C2000 real-time MCU applications, a typical use case is using the ADC input multiplexer to scan through multiple channels in a sequence. If a converted channel has inadequate settling, the channel may be pulled towards the voltage of the previous conversion in the sequence. This occurs because the S+H voltage starts near the previously converted voltage and then settles towards (but does not reach) the applied voltage. This tendency for the previous conversion result in a sequence of conversions to affect the current conversion is called memory cross-talk. Memory cross-talk can generally be completely mitigated via appropriate settling design.

A situation where a shared sample and hold must settle back-and-forth between two different multiplexed input signals is illustrated in Figure 2-4.

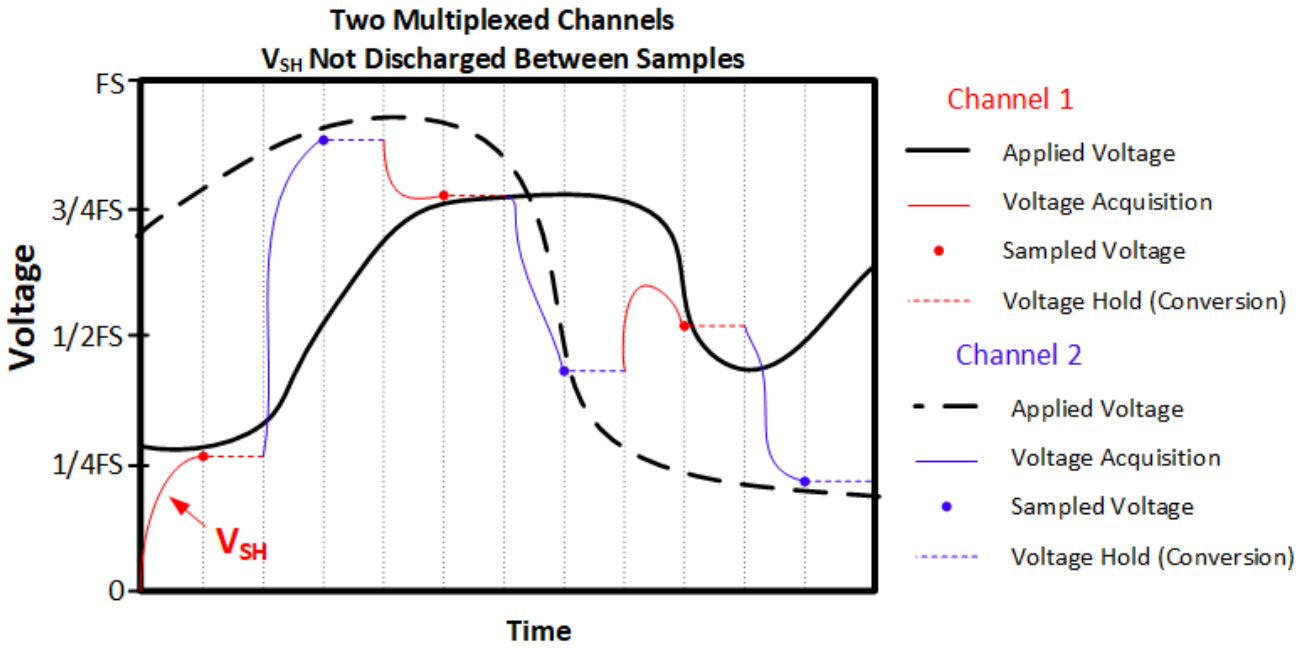

Figure 2-4. Sequence of Multiplexed Samples

Converter architectures that start with the S+H capacitor completely discharged generally do not experience significant memory cross-talk (but still experience input settling related distortion if the ADC driving circuits are not appropriate for the allocated acquisition time).

2.2.3 Accuracy

The errors introduced by inadequate input settling generally can not be calibrated out or reduced via oversampling and averaging. Therefore, applications that are concerned with absolute sampling accuracy also need to ensure proper ADC input settling even if the sensed input signal is low-frequency or even DC.

2.3 C2000 ADC Architecture

C2000 real-time MCU ADCs will generally start with the S+H capacitor pre-charged to a voltage close to the previous conversion result. The exception to this is for ADCs that support differential signaling, but which are operating in single-ended mode. In this case, the S+H capacitor will start discharged when the previous conversion was from an even numbered channel and the current channel is an odd numbered channel, or vice-versa. For example, the S+H capacitor will start discharged if channel A4 is being sampled after A3 (or vice-versa) but will start close to the previously converted voltage when sampling channel A4 after channel A2 or channel A1 after channel A3.

3 Problem Statement

The following sections present and analyze a circuit that exhibits issues with S+H settling performance. While this signal conditioning circuit could certainly achieve good settling performance through a hardware re-design, there are a number a situations that could lead to an engineer needing to deal with such a circuit as-is. These include additional design constraints preventing further optimization (such as cost or printed circuit board area constraints preventing additional circuits, such as an op-amp, from being added) or needing to interface with an existing circuit that cannot support significant hardware changes.

3.1 Example System

Consider a system with the following ADC parameters:

- F280049 C2000 MCU operating in 12-bit external reference mode
- The ADCs are driven by an 100kHz ePWM trigger source (ePWM1 SOCA)
- Two ADCs are used: ADCA and ADCB
- Four voltages to are converted: V1 through V4
- V1 and V2 are assigned to ADCA and V3 and V4 are assigned to ADCB

The timing diagram of the example system can be seen in Figure 3-1.

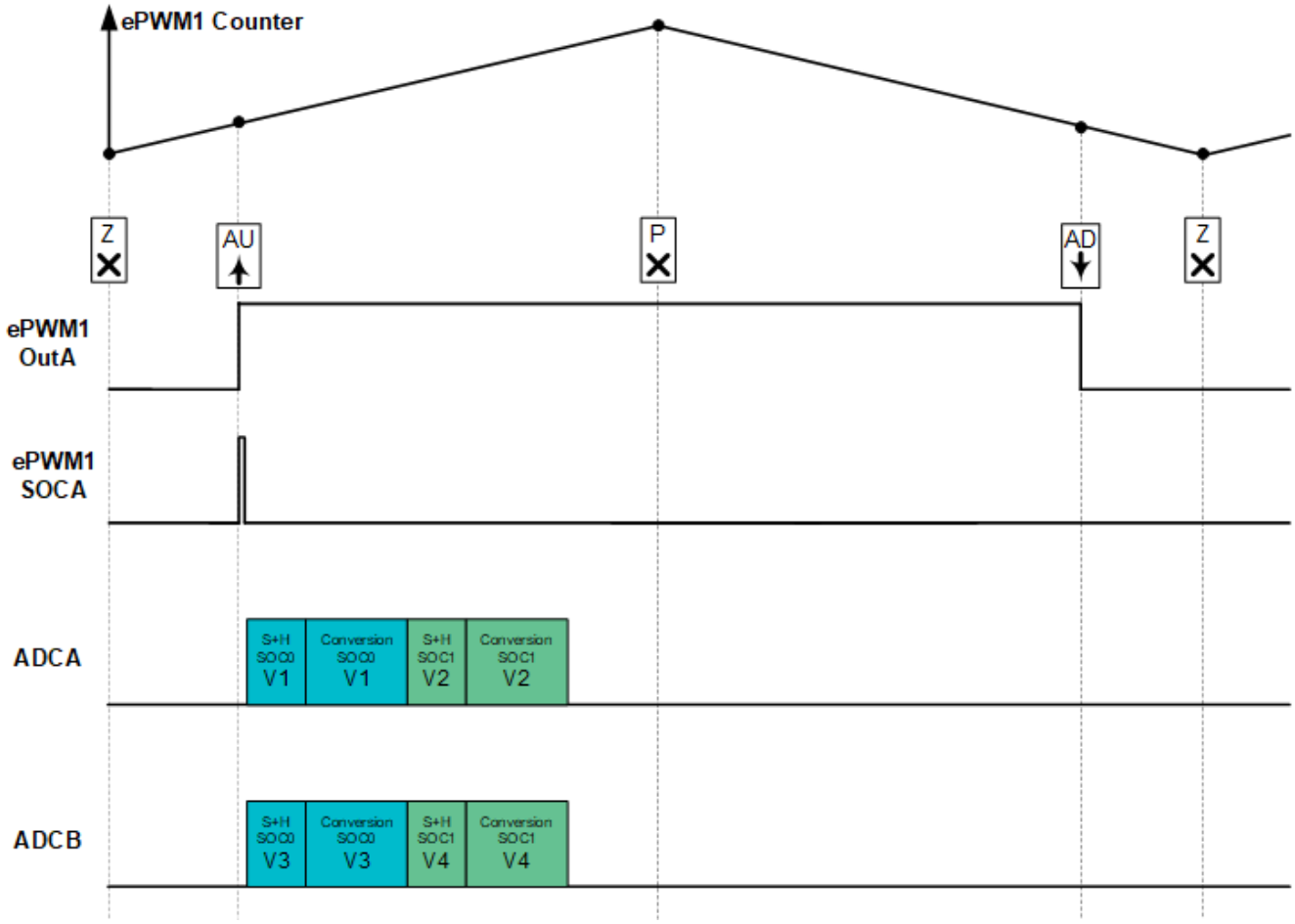

Figure 3-1. Example System Timings

Furthermore, consider that V2 is a high-voltage sensing circuit that uses a voltage divider comprised of a 1MΩ and 7.5kΩ resistors to scale down a 400 V signal to a 3.0 V range compatible with the ADC. The voltage divider directly drives the ADC input (no op-amp is used to buffer in the input) and the external capacitor has been selected to be 250 pF. The schematic for this circuit can be seen in Figure 3-2.

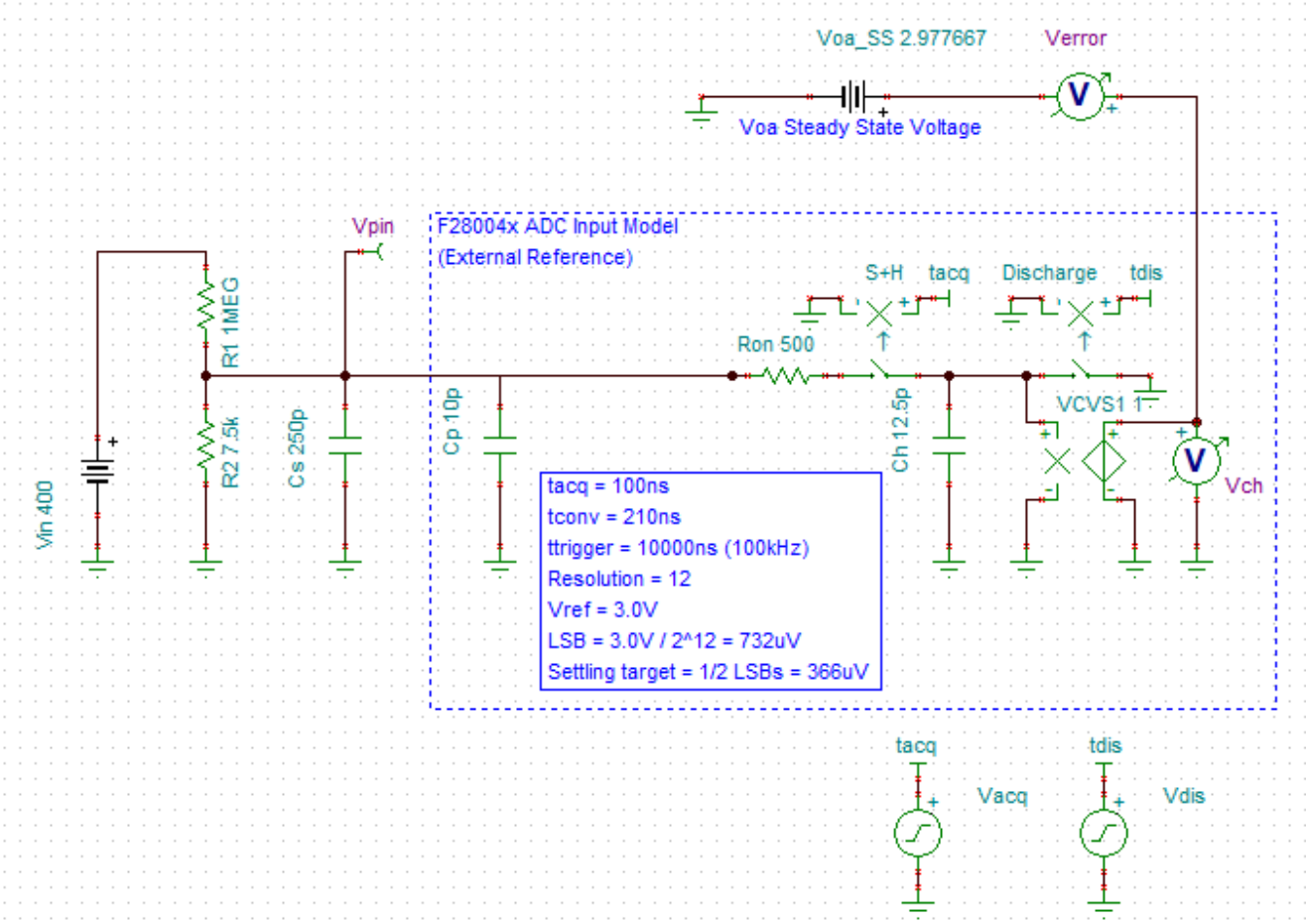

Figure 3-2. Example System Schematic for V2

As the subsequent sections will demonstrate, it would be difficult to select a S+H duration that is long enough for this circuit to achieve acceptable settling performance and the sample rate of 100 kHz rate is also too high to support a charge sharing input design. This will necessitate the alternate methods for mitigating memory cross-talk considered later in this document.

Note

An easy way to reduce the source impedance (and thus improve the settling time) would be to proportionally reduce both resistors in the voltage divider. However, this will increase the static bias current flowing through the resistor divider. For applications with aggressive power consumption requirements, the desire to not increase this bias current may be one reason why the source impedance is kept high.

3.2 S+H Settling Analysis

As presented in *Charge-Sharing Driving Circuits for C2000 ADCs*, an approximation of the required settling time can be determined using an RC settling model. The time constant for the model is given by the equation:

$$\tau = (Rs + Ron) \cdot Ch + Rs \cdot (Cs + Cp) \quad (1)$$

And the number of time constants needed is given by the equation:

$$k = ln\left(\frac{2^n}{settling\ error}\right) - ln\left(\frac{C_S + C_P}{CH}\right) \quad (2)$$

So the total S+H time should be set to approximately:

$$t = k \cdot \tau \quad (3)$$

Where the following parameters are provided by the ADC input model in the device-specific data manual:

- n = ADC resolution (in bits)
- R_{ON} = ADC sampling switch resistance (in Ohms)
- C_H = ADC sampling capacitor (in pF)
- C_P = ADC channel parasitic input capacitance (in pF)

The following parameters are dependent on the application design:

- Settling error = tolerable settling error (in LSBs)
- R_s = ADC driving circuit source impedance (in Ω)
- C_S = capacitance on ADC input pin (in pF)

Table 3-1 shows the settling time calculation using the values from the F280049 data manual and 250pF for C_S. R_s is set to 7444Ω, which is the effective impedance of the voltage divider formed by the 1MΩ and 7.5 kΩ resistors (1MΩ || 7.5 kΩ).

Table 3-1. Settling Time for V2 Circuit

Parameter	Example 1
C_S	250 pF
C_H	12.5 pF
C_p	10 pF
R_s	7444Ω
R_{ON}	500Ω
n	12 bits
settling error	0.5 LSBs
τ	1.96 µs
k	6.015
Settling time	11.8 µs

This analysis indicates that a settling time of 11.8 µs would be needed to achieve full 12-bit settling performance. This is, unfortunately, much longer than the maximum S+H time that the F280049 device's ADCs can be configured for. Furthermore, this is even longer than the 10 µs triggering period resulting from the 100 kHz ePWM trigger source. The latency resulting from such a long S+H window is also likely to be quite detrimental to the control system's performance. Overall, it will not be possible to get good settling performance just by configuring the correct S+H time.

Figure 3-3 shows the results of simulating Figure 3-2 while setting the S+H time to 100 ns. This time is much less than the 11.8 µs estimated to give full 12-bit settling performance, but represents what might be reasonably configured given the 100 kHz sample rate and a desire for low latency sampling. The simulation shows 131 mV of settling error after the 100 ns S+H window, which is about 4% of the ADC's 3.0 V range. This is quite poor performance considering that an ideal 12-bit ADC is capable of resolving steps of about 0.02% of the ADC's full-scale range.

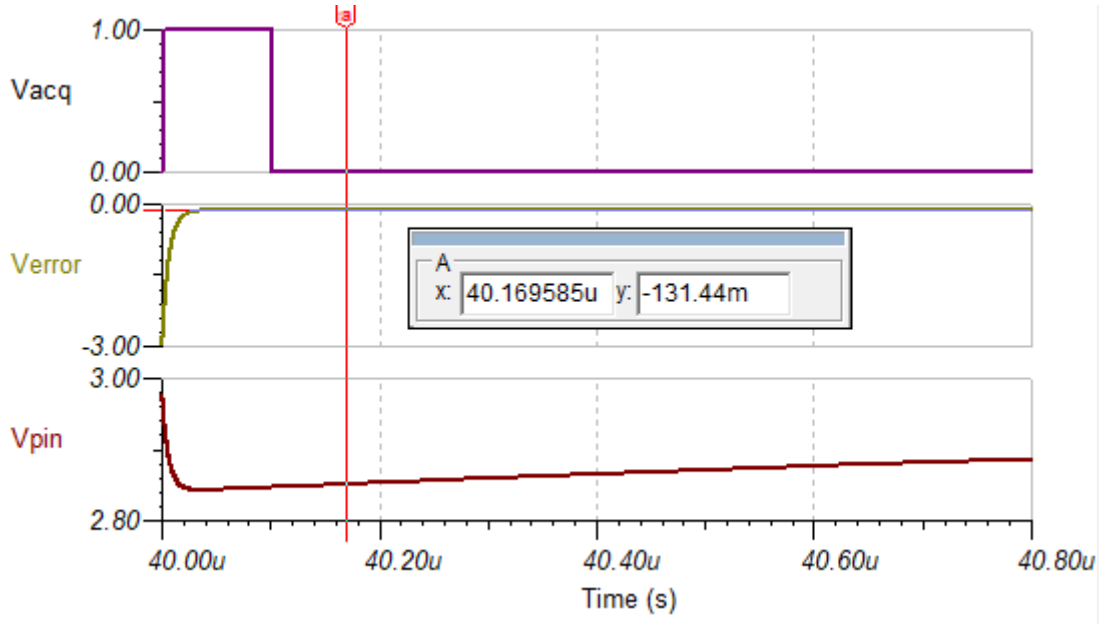

Figure 3-3. Settling Simulation for V2 Circuit With 100 ns S+H

3.3 Charge-Sharing Analysis

The previous section demonstrated that using an aggressive S+H window with this example circuit results in significant settling error. Another option would be to investigate whether increasing the external capacitor value on the V2 pin and using a charge sharing design might provide a feasible alternative design with only a minor hardware change.

As presented in *Charge-Sharing Driving Circuits for C2000 ADCs*, the external capacitor in a charge-sharing application should be selected to be:

$$C^S = \left(2^{N+2} \cdot C_H\right) - C_p \tag{4}$$

In this case, C_s would then be selected to be approximately 200 nF since C_H is 12.5 pF and N is 12-bits.

Given this capacitor value, it is possible to calculate the maximum acceptable sample rate using the formula shown in Equation 5:

$$f_s \leq 1/(0.7 \cdot R_S C_S) \tag{5}$$

Since C_s is 200 nF and R_s is 7444Ω (the parallel combination of the 1MΩ and 7.5kΩ resistors in the voltage divider) the sampling rate on this channel, f_s, should be kept below about 960Hz. Since the sample rate is fixed at 100 kHz, the charge sharing input design also suffers from a large disconnect between the operating parameters for optimum performance and the actual parameters that the circuit is using.

3.4 Problem Summary

Due to the high effective source impedance of the voltage divider along with the design not using an op-amp buffer to drive the ADC channel, voltage V2 will experience significant S+H settling error. Since the system uses a multiplexed sampling scheme where ADCA samples voltage V1 before V2 each time a trigger is received, this settling error will manifest as memory cross-talk from V1 to V2 as show in Figure 3-4. This means that when V1 is lower than V2, the sample results from V2 will be pulled lower and when V1 is higher than V2, the results of V2 will be pulled higher. Furthermore, the magnitude of this error is expected to be about 130 mV if V1 and V2 are at opposite extremes of the ADC's input range (as shown by the simulation in Figure 3-3).

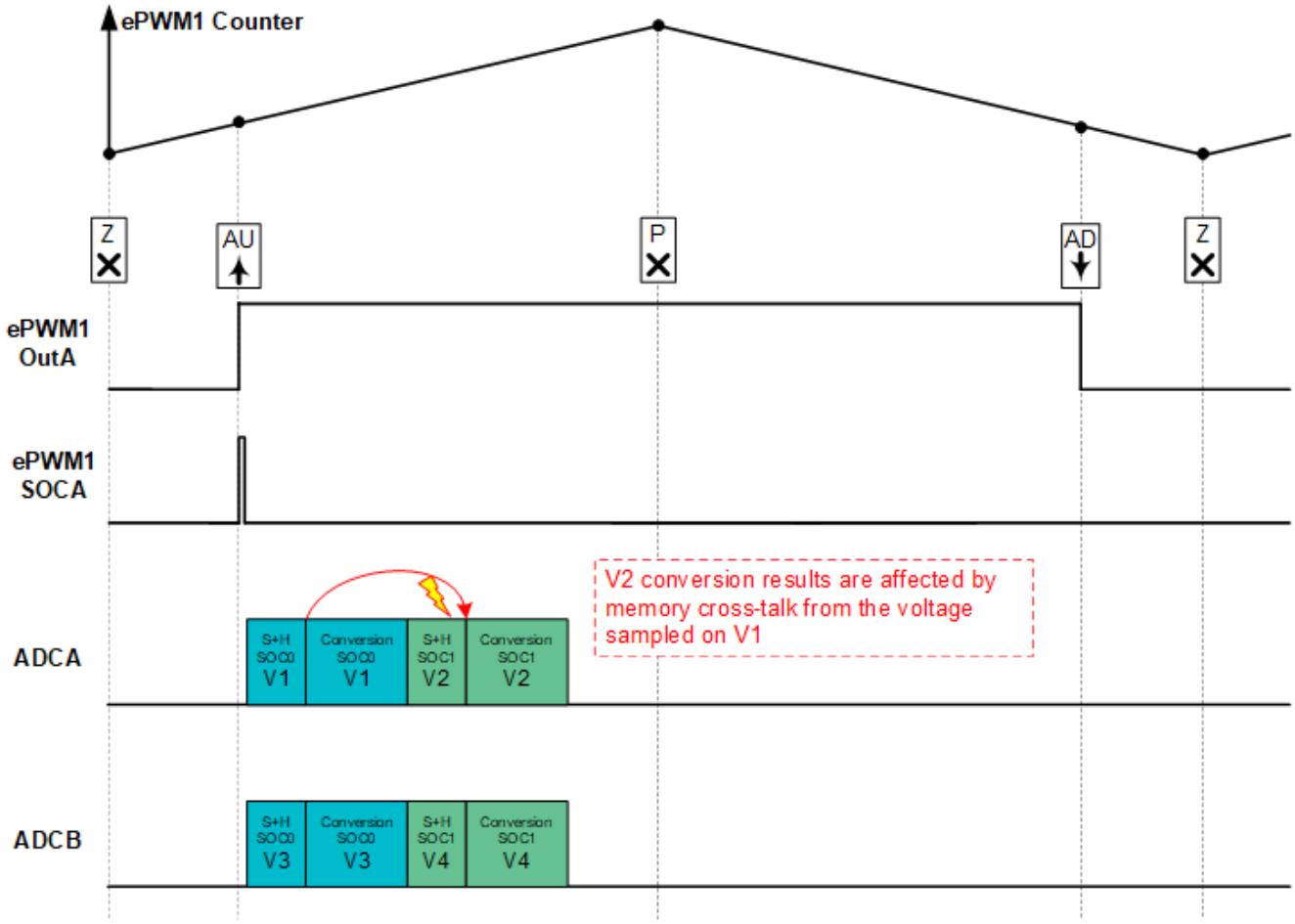

Figure 3-4. Example System Memory Cross-talk

Generally, since V1 and V2 independent signals being sampled by the system, the memory cross-talk error induced in V2 will vary unpredictably. This will likely result in poor system performance and instability. Ideally, the signal conditioning circuits driving V2 would be redesigned to ensure good settling performance, but this is not always feasible due to additional system constraints or working with a fixed existing system. Section 4 and Section 5 present methods for transforming the memory cross-talk error into a form that is more systematic and predictable.

4 Dedicated ADC Sampling

Memory cross-talk between signals occurs when a signal with poor settling performance follows another signal in a multiplexed sampling sequence. In systems with multiple ADCs and a very small number of signals with poor settling performance, the affected signals can be partitioned onto their own dedicated ADC modules. This replaces the memory cross-talk from an unrelated signal with memory cross-talk from the previously sampled value of the same signal.

The following sections will describe dedicated ADC sampling strategy in more detail, discuss the advantages and limitations of this method, and provide instructions for evaluating the signal performance via simulation.

4.1 Dedicated ADC Concept

In the system presented in Section 3.1, two ADCs are used to sample four voltages, only one of which is specified to have poor input settling performance (voltage V2). Allocating two signals to each of the two ADCs results in the shortest overall latency to process the four voltages. However, if the sampling scheme is rearranged as shown in Figure 4-1, then the signal with poor settling performance (voltage V2) receives a dedicated ADC.

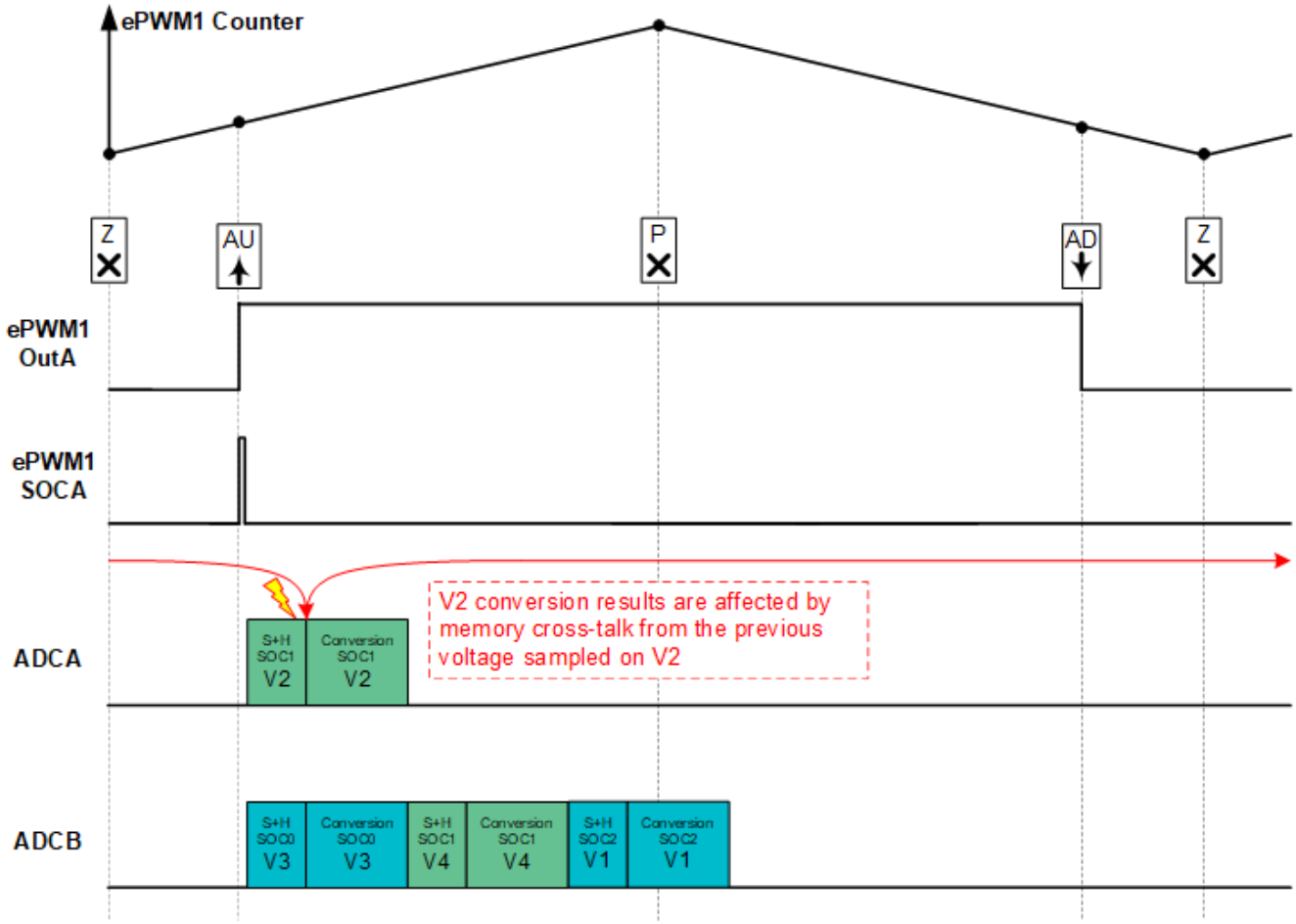

Figure 4-1. Timings for Dedicated ADC Method

The main benefit of isolating the affected signal onto a single ADC is that the memory cross-talk error now originates from the previously sampled value of the same signal. If the signal is not moving very fast, the S+H circuit will start with a value close to the current input voltage. This greatly reduces the distance the input has to settle, resulting in acceptable settling performance. However, if the input experiences a large step in the input voltage, it may take several samples for the ADC to catch up with the new value on the input. Thus, a dedicated ADC strategy is generally more appropriate for slow moving or DC input signals. However, even in the case of faster signals, the memory cross-talk effect at least becomes systematic, which is usually an improvement compared with a memory error coupling into the signal in question from an unrelated signal.

Of course, dedicating an ADC to sample a single input signal results in increased latency in processing the remaining signals in the application. For instance, after re-arranging the signals as shown in Figure 4-1, the latency to produce the final ADC result has increased by 50%! (assuming all conversions are configured to take the ADC equal amounts of time to process).

4.2 Settling Mechanism for Dedicated ADC

Figure 4-2 shows the setup for a simulation of the first few periods of settling for the example system rearranged as described in Dedicated ADC Concept. A 100ns S+H window will be used, which is much shorter than the time previously determined to be necessary for good settling under normal conditions (11.8 µs). Note that the portion of the input model which discharges the S+H capacitor between samples has been removed, so C_H will retain its charge between samples. Finally, note that the simulation is configured such that the initial voltage condition on C_H is 0 V while the voltage divider output is a voltage near full-scale (3.0 V). This will result in a simulation equivalent to a full-scale input setup.

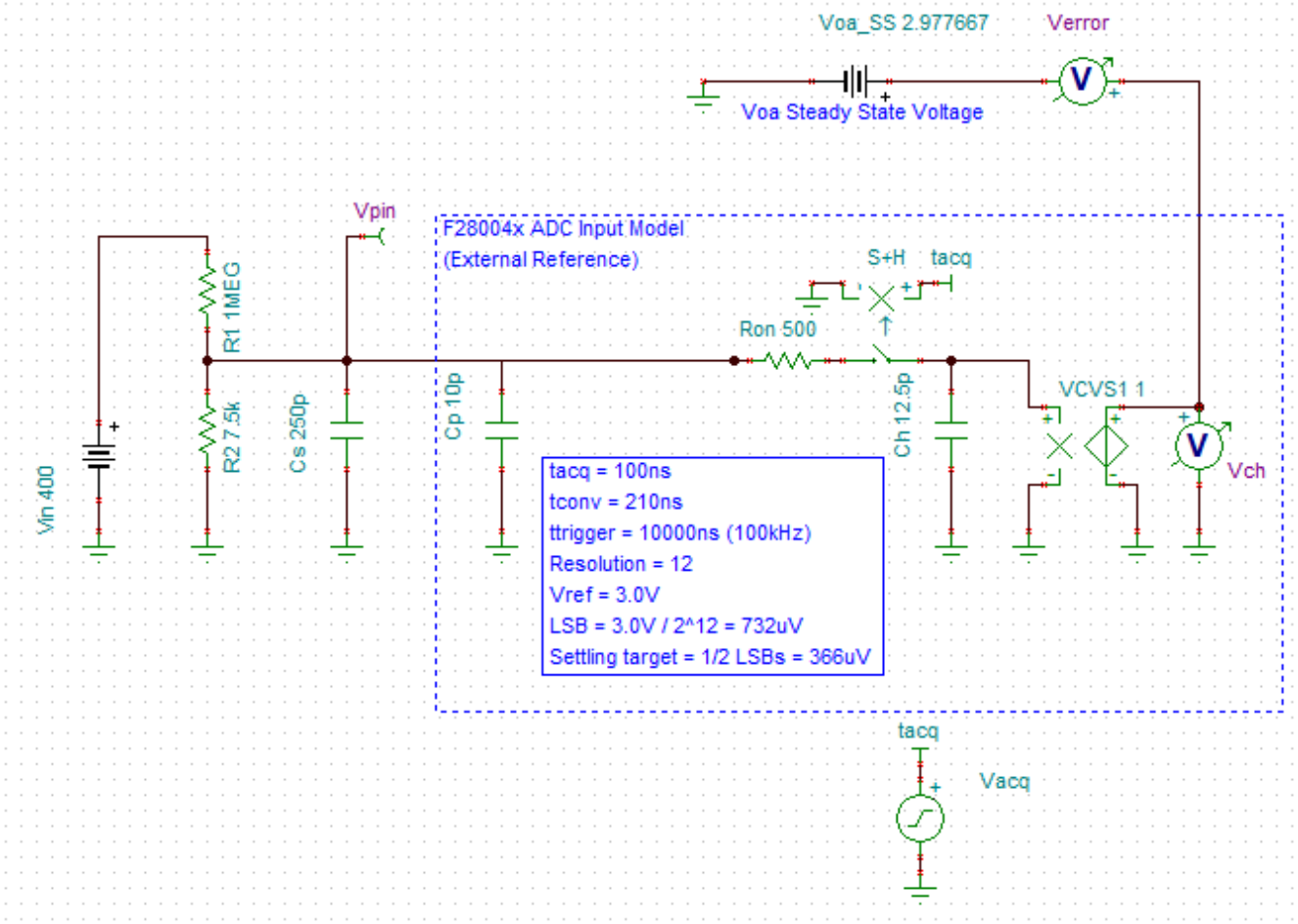

Figure 4-2. Simulation Schematic for V2 on Dedicated ADC

Running a transient simulation for 30µs gives the results as shown in figure Figure 4-3. The settling error after the first sample is about 100 mV, which is then reduced to 5.4 mV and then 0.27 mV after the second and third samples, respectively. Since 1/2 LSB settling with a 3.0 V ADC range is about 0.37 mV, the system did eventually achieve good settling in response to a full-scale step response! However, this took three sample periods (30 µs) and assumed that the signal remained stationary after the step response.

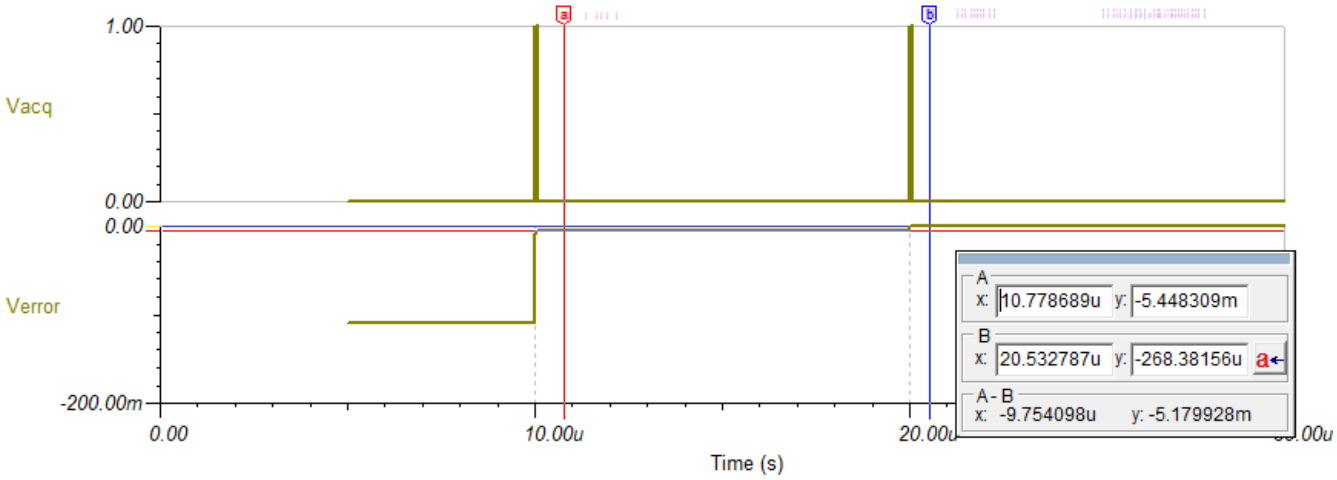

Figure 4-3. Settling Simulation for V2 Circuit on Dedicated ADC

For a review of the structure of this simulation and the methods needed to run the simulation, see *ADC Input Circuit Evaluation for C2000 MCUs*.

What is quite interesting about these results is that 11.8 µs of settling time was predicted as necessary for this circuit topology, but the ADC only had the S+H window open for three bursts of 100 ns, resulting in 300 ns of total sampling time. How then did the circuit achieve good settling?

For each sample, note that the error scales down by roughly a factor of 20. This corresponds to the ratio of the external ADC input capacitor, C_S (250 pF in this case), to the internal ADC S+H capacitor, C_H (12.5 pF in this case). When the S+H first opens, these two capacitors quickly equalize, charging C_H 95% of the way towards the input voltage present on C_S.

4.3 Design Flow for Dedicated ADC

The previous section demonstrate that almost all of the charging or discharging of the S+H capacitor C_H in a dedicated ADC configuration comes from charge equalization, so a S+H value near the minimum value supported by the ADC can be used (since the charge equalization occurs quickly at the beginning of the S+H window). Furthermore, re-running the simulation in Figure 4-3, but with progressively increasing values of the capacitor on the ADC pin will result in progressively better settling performance at each sampling step (not shown). Therefore, it is generally desirable to maximize the size of C_S (if this can be changed in the system hardware) as this will maximize the settling performance. However, note that the source impedance, R_s, and the ADC pin capacitance, C_s form a low-pass filter. Therefore, the size of C_s will need to be limited to ensure the input circuit has sufficient bandwidth to not distort the signal of interest.

These design decisions are summarized in the following list:

- Isolate the signal affected by memory cross-talk to a dedicated ADC
- S+H window duration can be set to the minimum allowed by the ADC (or any other valid convenient value)
- If possible, modify the circuit to maximize C_s (the capacitance on the ADC pin) being mindful that the input circuit bandwidth (BW_{RsCs}) needs to be large enough to allow the circuit to sense the frequencies of interest in the input signal.

$$BW_{RsCs} = 1/(2\pi \cdot C_S \cdot R_S) \quad (6)$$

- (Optional) Simulate the settling performance at frequencies of interest (see next section)

4.4 Simulating Settling Performance for a Dedicated ADC Circuit

In Section 4.2, a simple simulation was presented for the response of a dedicated ADC design to a single input step. However, simulation can also be used to evaluate the settling performance using an AC source. This allows for investigation of how the settling performance changes with increasing input frequency.

Before proceeding, it may be useful to first review the methods for performing an AC Input Simulation presented in the TI Precision Labs video: Final SAR ADC Drive Simulations.

Figure 4-4 shows the simulation setup for the AC simulation. The 400 V DC input source has been replaced with a 400 Vpp sine input. On the top, a duplicate of the original circuit, minus the sampling, has been added. This is necessary to generate a reference waveform that the sampled voltage can be compared to. If the input source is used directly for comparison the phase delay from the various R-C components will obscure the settling error.

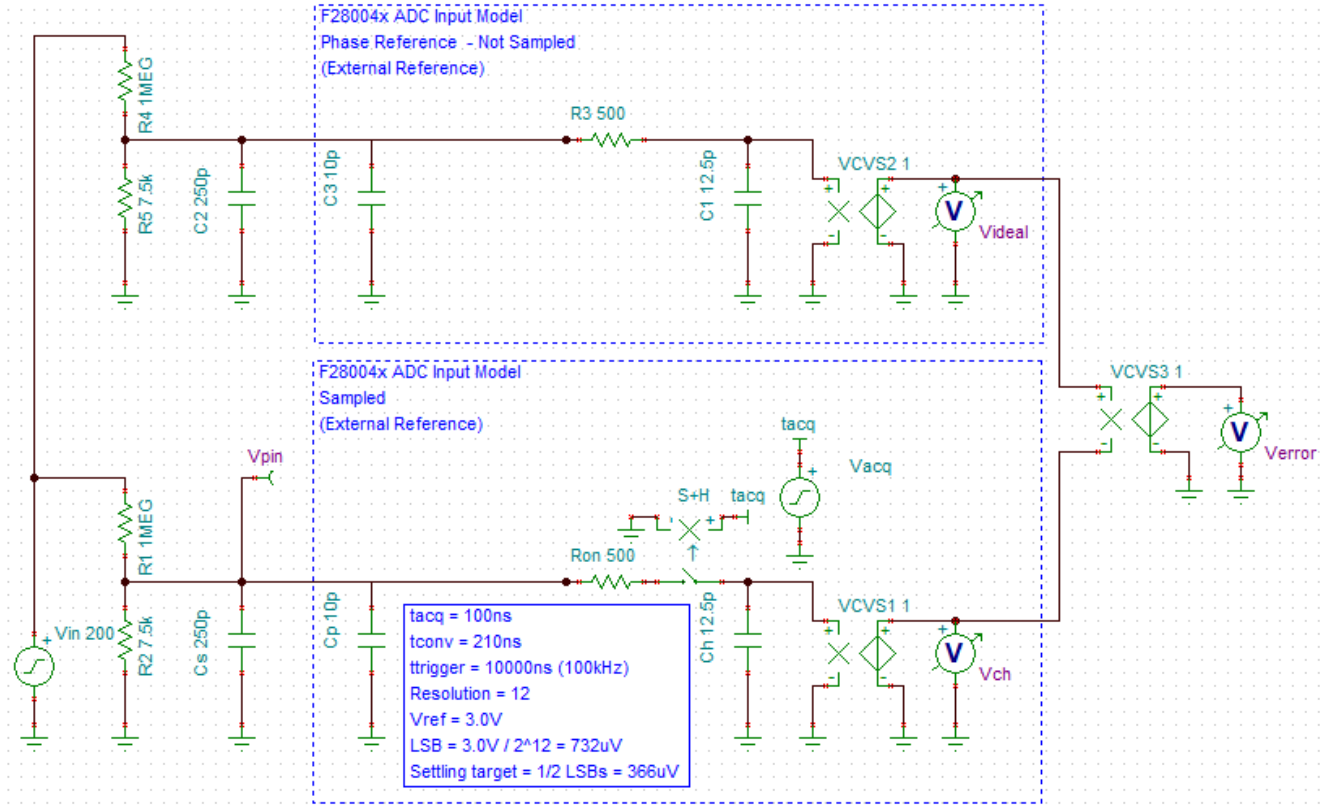

Figure 4-4. Schematic for AC Simulation for Dedicated ADC

Figure 4-5 shows the results of simulating a 60Hz AC input over many samples over 1 ms. Selecting an arbitrary sample while the sine wave is rising and examining the error waveform shows a settling error of about 0.2 mV right at the end of the S+H window. This level of settling error indicates the circuit is achieving good settling performance with a 60Hz input (1/2 LSB with a 3.0 V range is 0.37 mV).

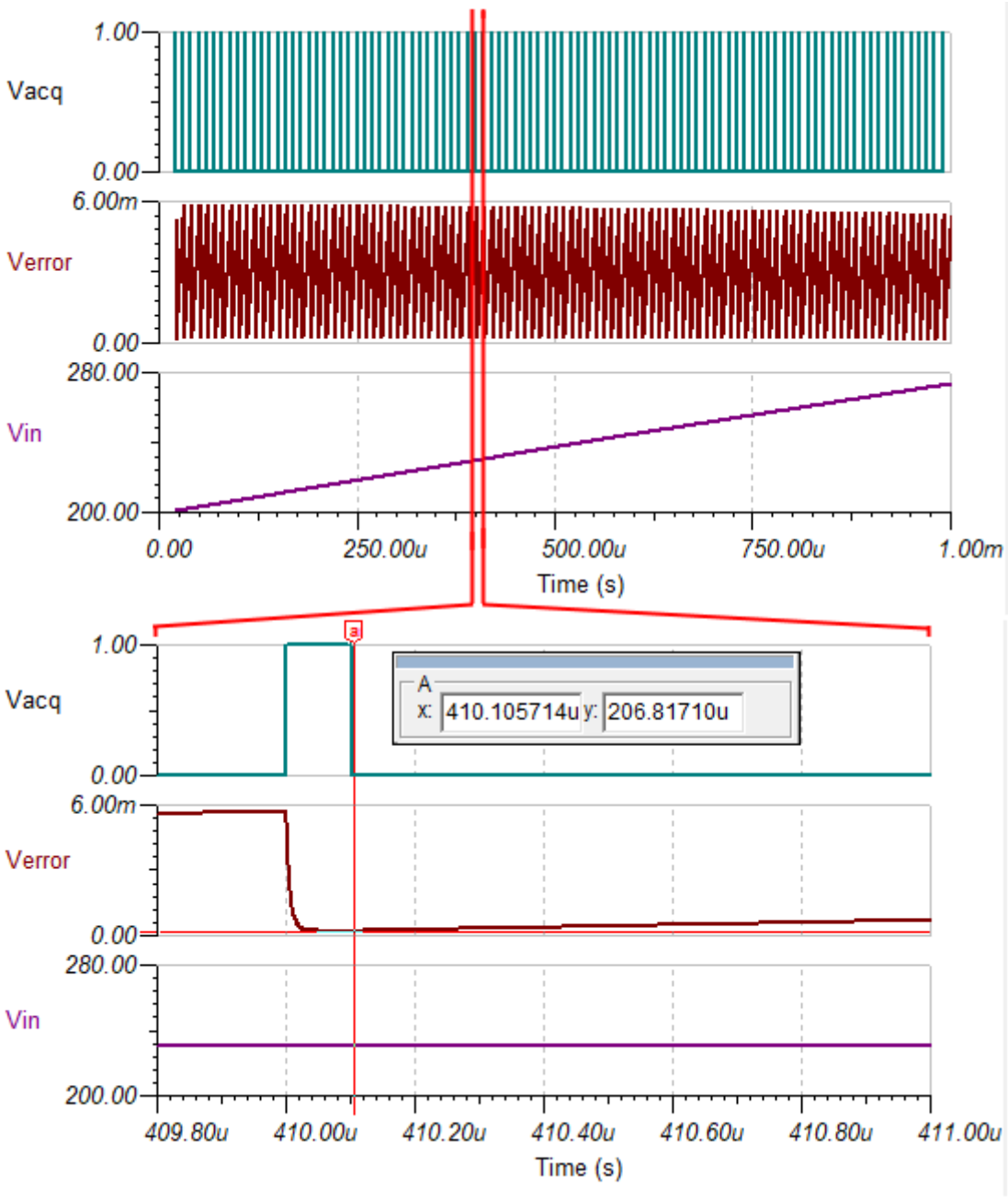

Figure 4-5. Settling Simulation for 60Hz AC Input

Compared to the 60Hz input, Figure 4-6 shows the results of instead simulating with a 6kHz input. Taking the error at the end of the S+H period while the waveform is decreasing shows a settling error of approximately 19 mV, so the circuit does not perform very well when trying to track a much faster input. Note, however, that increasing the external ADC pin capacitor, C_S, improves the settling performance at higher speeds. Care needs to be taken to ensure C_S is not made so large that the 6 kHz input is attenuated by the low-pass filter formed by C_S and R_S.

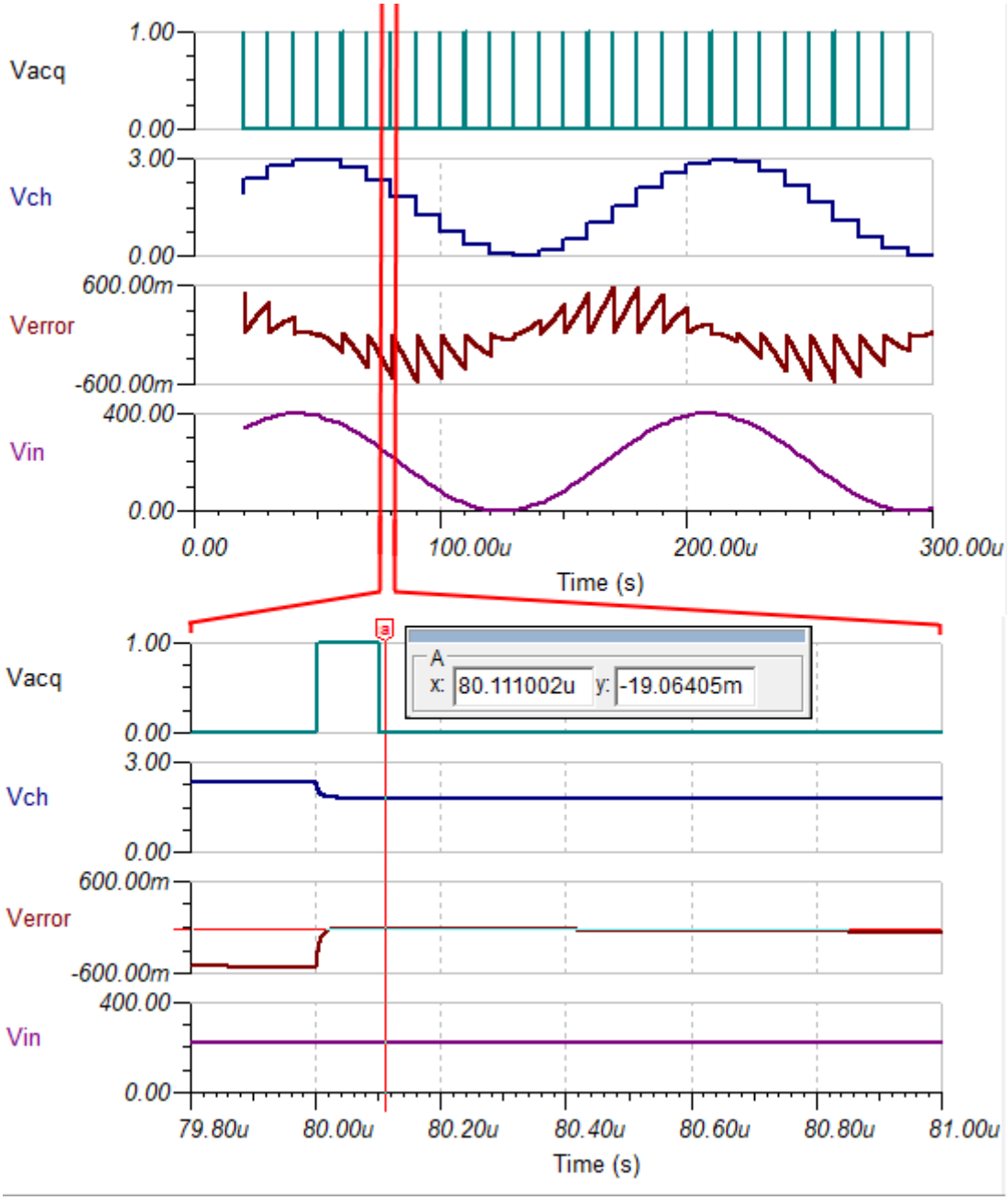

Figure 4-6. Settling Simulation for 6kHz AC Input

In addition to the settling error increasing with increasing ADC input frequency, another important property of the settling error to note is that faster moving portions of the input signal will have increased error when compared to slower moving portions of the signal. Because of this, the settling error will not equally affect all portions of the input sine wave, resulting in the sampled waveform being distorted.

5 Pre-Sampling VREFLO

Another strategy for mitigating memory cross-talk is to add a sample of a fixed voltage immediately before sampling the channel which exhibits poor settling performance. This will result in settling error that is at least deterministic.

Since almost all C2000 based MCUs include an internal connection to VREFLO (0V) in the ADC channel mux, sampling VREFLO is a convenient and effective way to sample a fixed voltage. Therefore, adding sampling slots between channels that are experiencing memory cross-talk and sampling an internal VREFLO connection in these slots is a good memory cross-talk mitigation strategy on C2000 MCU devices (assuming that re-designing the ADC driving circuit for adequate settling performance is not feasible).

The subsequent sections will further describe how and why to insert samples of VREFLO into the sampling slots between normal conversions to help mitigate memory cross-talk. Quantification of the systematic settling error that results will also be discussed as well as ways to further mitigate this resulting error (gain error compensation).

5.1 VREFLO Sampling Concept

In Section 3.1, an example system that sampled four voltages was presented. One of these voltages, V2, was specified to have poor settling performance (due to excessively high source impedance from using a voltage divider to drive the ADC input directly). Figure 5-1 shows one possible re-organization of the sampling scheme where a sample of VREFLO (zero-scale) has been added before the channel in question. The result of this is to replace memory cross-talk from voltage V1 (which is an independent signal that could take on any voltage at any time) with memory cross-talk from VREFLO (that reliably produces cross-talk that pulls V2 towards exactly 0 V).

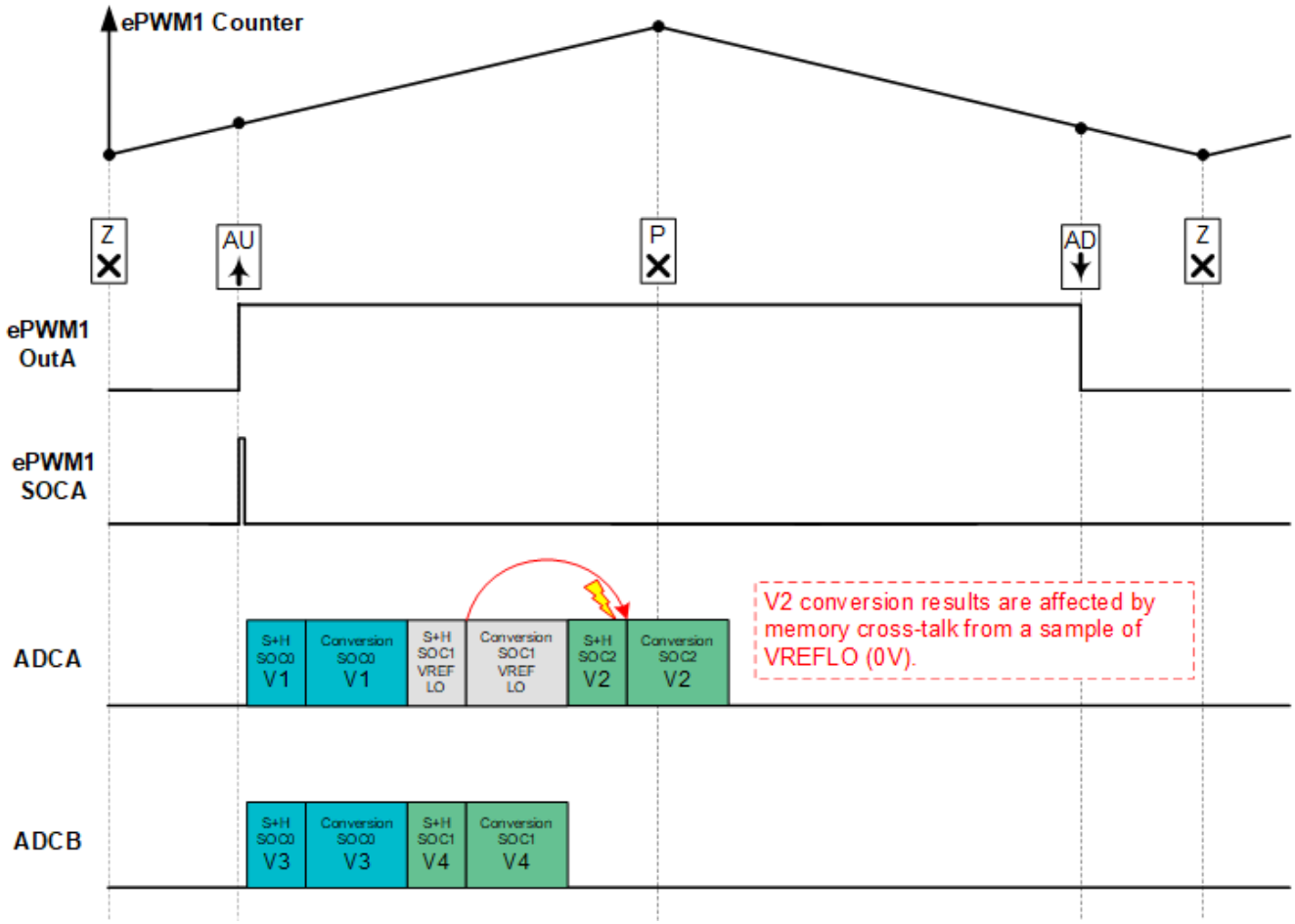

Figure 5-1. Timings for VREFLO Sampling Method

Unlike the single-ADC method, it is possible to apply the pre-sampling VREFLO methodology to multiple signals on the same ADC. This method is only limited by the number spare start-of-conversion (SOC) structures available in the ADC wrapper to sequence the extra conversions (no external pins are needed) and the amount of additional latency spent processing the VREFLO samples that the application can tolerate. Later sections will also show that a light post-processing of the ADC results is also desirable. Performing this operation will add some additional latency that will also need to be absorbed by the application.

5.2 Properties of VREFLO Sampling Method Error

Simulating a circuit with memory cross-talk from VREFLO requires minimal changes to the standard ADC input settling simulation presented in *ADC Input Circuit Evaluation for C2000 MCUs*. The default setup already forces a positive voltage while the S+H capacitor, CH, starts from an initial condition of 0 V.

A simulation setup for measuring the VREFLO settling error is shown in Figure 5-2. A 200 ns S+H time is used. Table 5-1 shows the result of simulating the sampled voltage for a range of input voltages. As can be seen in the rightmost column "Sampled Voltage Error (%)", the sampled voltage including the VREFLO cross-talk error manifests as a linear scaling error (gain error).

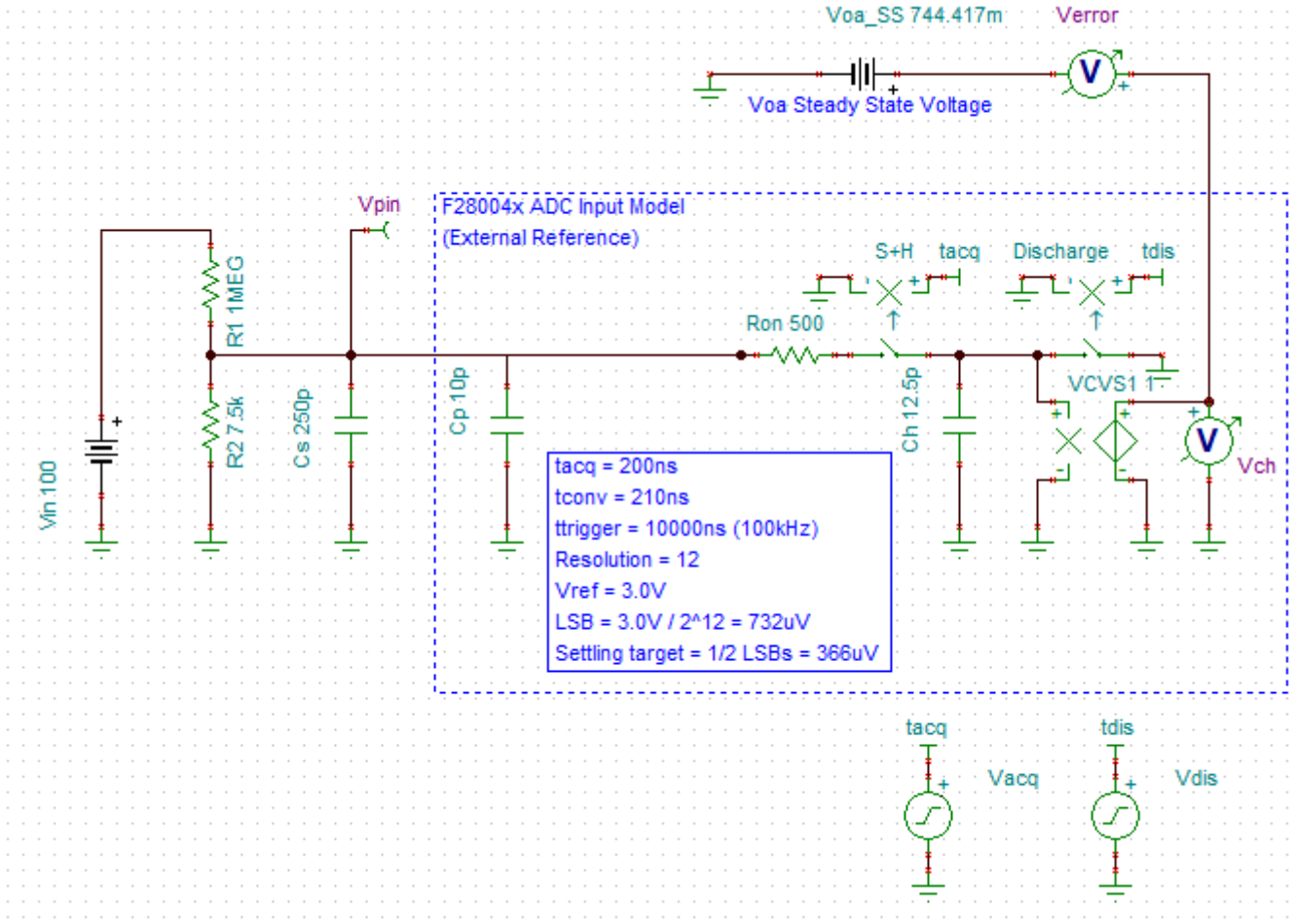

Figure 5-2. Schematic for VREFLO Sampling Simulation

Table 5-1. Sampled Voltage vs. Vin With VREFLO Cross-Talk

Vin	Vpin	Sampled Voltage	Sampled Voltage Error (%)
400 V	2.978	2.853	-4.39
300 V	2.233	2.139	-4.39
200 V	1.489	1.426	-4.39
100 V	0.744	0.713	-4.39

5.3 Gain Error Compensation

Sampling VREFLO before a conversion with inadequate settling performance transforms the memory cross-talk from another signal (which could be unpredictable) to a constant and predictable scaling factor as shown in Equation 7. However, the resulting gain error in that example was over 4%. This is quite large and will likely be the dominant error source in the system when determining the overall accuracy for the conversion of that signal.

To mitigate the gain error created by the VREFLO sampling method, the user can post-process the ADC results. This entails performing a simple scaling (multiplication) operation on the raw ADC results to correct for the gain error introduced. This can be done in the CPU ISR or CLA task that reads the ADC results by performing the simple operation:

$$V_i' = V_i \cdot C_i \tag{7}$$

Where:

- V_i = the raw ADC result affected by the VREFLO sampling
- V_i' = the compensated ADC result
- C_i = the compensation coefficient for channel i

For example, if the relative error caused by the settling error is -4.39%, C_i would be (1 - (-0.0439)) = 1.0439 and if the raw ADC result was 2199 the compensated ADC result would be 2199·1.0439 = 2295.5.

5.3.1 Methods for Determining Compensation Coefficients

For each channel to be compensated for gain error arising from the VREFLO sampling method, a calibration coefficient needs to be determined. There are two potential methods to determine these coefficients :

- Determine the coefficient via simulation

  Coefficients can be determined via input settling simulation similar to that performed in Figure 5-2. Once the settling error is determined, the coefficient can be calculated using the formula

- $$C_i = V_{in} / (V_{in} - settling\ error) \tag{8}$$

- Determine the coefficient via in-system calibration

Instead of measuring the input and output voltages in simulation, the settling induced gain error can be directly measured in the end application. A known DC voltage is applied to the channel and then sampled through the ADC. Any deviation from the expected value is assumed to be memory cross-talk induced settling error. Calculation of the coefficient proceeds using Equation 8.

In cases where system gain calibration was already planned for the end application, compensation for the gain error caused by the VREFLO sampling method can be absorbed into the existing calibration scheme (only one coefficient is needed).

5.4 VREFLO Sampling Design Flow

When configuring the ADC to use VREFLO samples to mitigate memory cross-talk, several decisions need to be made. With respect to the selected S+H duration, two values need to be selected: the S+H duration for the channel sampling VREFLO and the S+H duration for the channel sampling the signal of interest. The channel sampling VREFLO can use the minimum S+H window allowed by the ADC, but some care needs to be taken to select the S+H for the channel of interest. The more settling error the selected S+H produces, the more gain error needs to be compensated out of the system. In general, the magnitude of this gain error should be kept to a reasonable level because a larger value will be more sensitive to component tolerances. Furthermore, the attenuation of the input signal caused by the gain error could cause less of the ADC input range to be used by the application, reducing dynamic range of the signal. For these reasons, it is recommended that the S+H be selected to keep the gain error less than about 5% of the ADC full-scale range (even though calibration should be able to remove most of this error).

With a method to select the S+H duration determined, the user has all the information needed to design a system using the VREFLO pre-sampling method. They should follow the following steps:

- Add additional SOCs to the ADC sampling sequence and configure those SOCs to sample the internal connection to VREFLO using the minimum S+H window duration allowable (or another convenient and valid S+H duration)
- Simulate the settling error with different S+H durations to determine a S+H duration that gives a settling error less than about 5% of the ADC full-scale range
- Determine the compensation coefficients for each channel benefiting from the VREFLO sampling method through either (1) simulation of the settling error or (2) direct measurement of the settling error. Once the settling error is determined, the coefficient, C_i, is given by the equation below, where V_{in} was the input voltage applied to the system or simulation.

$$C_i = V_{in} / (V_{in} - settling\ error) \quad (9)$$

- The use should add compensation code to their CPU ISR or CLA task to modify the raw ADC result using the following formula. V_i is the raw ADC result affected by the VREFLO sampling, V_i' is the compensated result, and C_i is the compensation coefficient determined in the previous step.

$$V_i' = V_i \cdot C_i \quad (10)$$

5.5 Discussion of VREFLO Sampling Sequences

As mentioned in Section 5.1, the VREFLO sampling method of mitigating memory cross-talk can be applied to multiple signals per-ADC (unlike the dedicated ADC method). There are some points to consider when setting up this sampling sequence. Figure 5-3 will be used as an example of a scheme where four signals spread across two ADCs all need to benefit from memory cross-talk mitigation using VREFLO sampling.

Consider first that even though the last SOC in each sequence (SCO3 = VREFLO) does not occur immediately before the first sample in the sequence (SOC0 = V1 or V3), for the sake of memory-cross talk it will be the sample influencing the first conversion in the sequence due to the periodic nature of the sampling. Because of this, appending VREFLO to the end of the sequence and making a channel affected by memory cross-talk the first sample in the sequence is an acceptable configuration.

Furthermore, consider where the trigger to the ISR should be placed in the sequence. Normally, the last SOC in the sequence would trigger the ISR to ensure that all ADC results are ready by the time the CPU enters the ISR. However, there is not need to read the VREFLO conversion results, so the ideal place to place the ISR trigger would be after SOC2, not after SOC3. Because of this, even through two signals are benefiting from VREFLO memory cross-talk mitigation in Figure 5-3, the latency only increases by the time needed to take one VREFLO sample, not two (the latency of the VREFLO conversion at the end of the sequence can be masked).

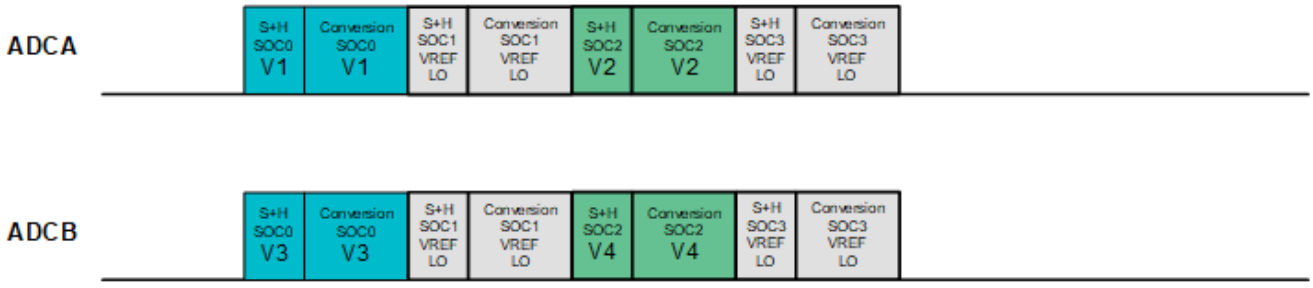

Figure 5-3. Timings for VREFLO Sampling for All Channels

6 Summary

Failing to design ADC input driving circuits in a real-time control application for proper S+H settling performance can lead to a variety of performance issues. Primary among these issues is memory cross-talk: an error where a sampled signal is affected by the previously sampled channel in the ADC sampling sequence. While the best course of action when this issue is discovered is to re-design the hardware circuits to achieve good settling performance, this may not always be feasible due to aggressive design constraints or working with an existing design that can't be significantly modified. In cases where hardware re-design is not viable, modifying the sampling sequence to move the affected channel to a dedicated ADC, or restructuring the sequence to sample VREFLO before the affected channel are possible options to help mitigate the memory cross-talk error.

The dedicated ADC method is simple and can use the minimum S+H duration to sample the channel of interest. However, this method may not be suitable for higher bandwidth signal and there will likely not be sufficient hardware resources to support many signals utilizing this method.

Adding a sample of VREFLO (0V) before the channel of interest is also a simple and convenient strategy to transform the memory cross-talk into a more deterministic form. It can be used for higher speed signals than the dedicated ADC method, but may not be able to use the minimum S+H duration. This method produces a significant amount of gain error, but this error can be easily calibrated out of the ADC results via an additional gain compensation scheme.

7 References

- TI Precision Labs - SAR ADC Input Driver Design
- Analog Engineer's Calculator
- TINA-TI™
- PSpice for TI
- Texas Instruments: *ADC Input Circuit Evaluation for C2000 MCUs*
- Texas Instruments: *Charge-Sharing Driving Circuits for C2000 ADCs*
- TI Precision Labs: Final SAR ADC Drive Simulations

8 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Revision * (June 2021) to Revision A (March 2023)	Page
• Updated the numbering format for tables, figures and cross-references throughout the document	3
• Updated Section 1.2.3	4

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265