TI Designs

Phase-Compensated, 8-Ch, Multiplexed Data Acquisition System for Power Automation Reference Design

TI Designs – Precision

TI Precision Designs are analog solutions created by TI's analog experts. Verified Designs offer the theory, component selection, simulation, complete PCB schematic and layout, bill of materials, and measured performance of useful circuits. The design also addresses circuit modifications that help to meet alternate design goals.

Design Resources

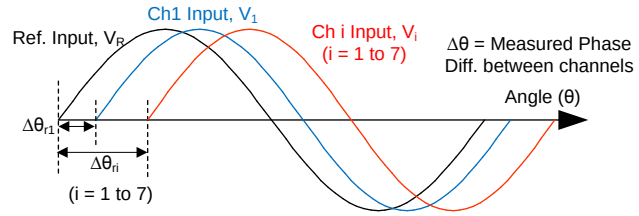
TIPD167	Tool Folder Containing Design Files
TINA-TI™	SPICE Simulator
ADS8688	Product Folder

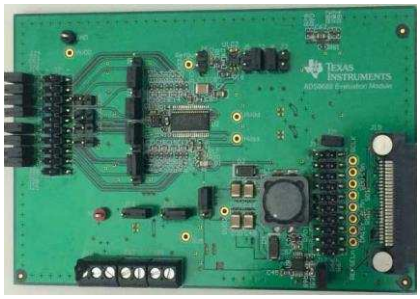
TI E2E™ Community

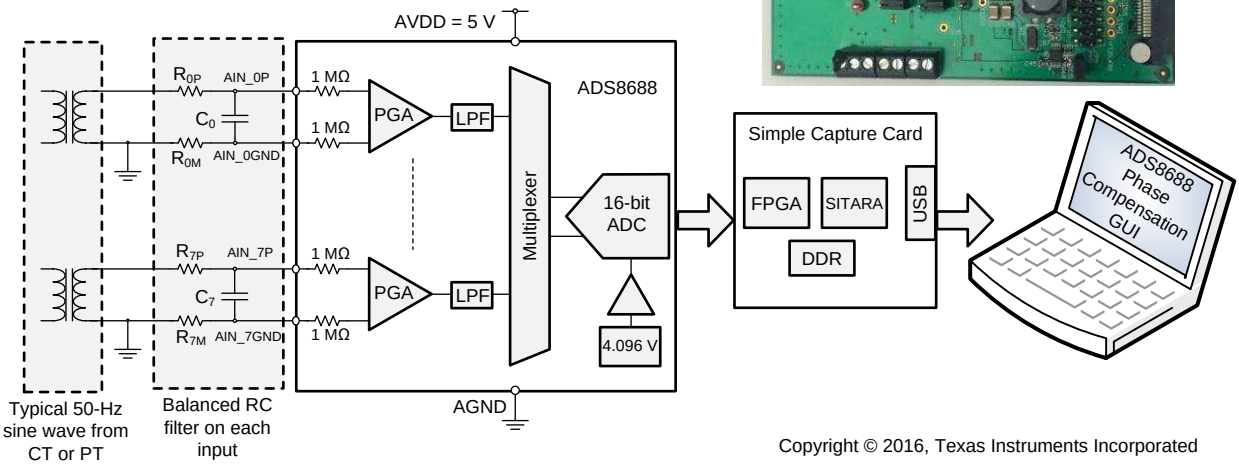
ASK Our E2E Experts
WEBENCH® Calculator Tools

Circuit Description

This design describes the importance of accurate phase measurement between electrical signals in power automation application. The circuit utilizes a 16-bit, multiplexed input (non-simultaneous sampling) successive approximation register (SAR) based analog-to-digital converter (ADC) with an integrated analog front-end circuit for the ease of system hardware design. This design compares the additional phase delay introduced between the input channels of a non-simultaneous sampling ADC with a simultaneous sampling ADC and also implements an optimum software solution to compensate for this phase delay.







1 Design Summary

The primary design objective is to accurately measure the phase and phase difference between the analog inputs in a multi-channel data acquisition system. When multiple input channels are sampled in a sequential manner as in a multiplexed ADC, an additional phase delay is introduced between the channels. The measured phase value of the signal includes this additional phase delay. Thus the phase measurements are not accurate. This additional phase delay is constant and can be compensated in application software. This design describes a software algorithm to compensate for the additional phase difference between the channels.

The key design requirements are as follows:

- Input signal: Single-ended sinusoidal input with an amplitude of V_{in}= ±10 V and typical f_{in} = 50 Hz
- Test equipment: AP2722 signal generator (audio precision) and ADS8688EVM with simple capture card

A multiplexed ADC with phase compensation can be used in place of the simultaneous sampling ADC for measuring phases in power automation applications based on the results achieved in this design. The performance of this design is summarized in Table 1 and Figure 1 depicts the measured phase difference.

Table 1. Comparison of Theoretical and Measured Performance

TEST CONDITION	THEORETICAL1 Δθ	MEASURED2 Δθ	COMPENSATED3 PHASE
Phase difference (two consecutive channels)	0.036º	0.036145º	0.000145º
Phase difference (farthest channels, Ch0 to Ch7)	0.252º	0.249964º	0.002036º

1. The theoretical phase difference introduced by multiplexing is calculated based on the formulas in Section 2.3.2.
2. A 50-Hz sine wave signal is connected to all channels of the ADC sampling at f_{ADC}= 500 kSPS. The measured phase value (before compensation) includes the phase difference as a result of the multiplexing structure.
3. The theoretical phase difference is subtracted from the measured phase to compensate for the phase difference as a result of the multiplexing structure.

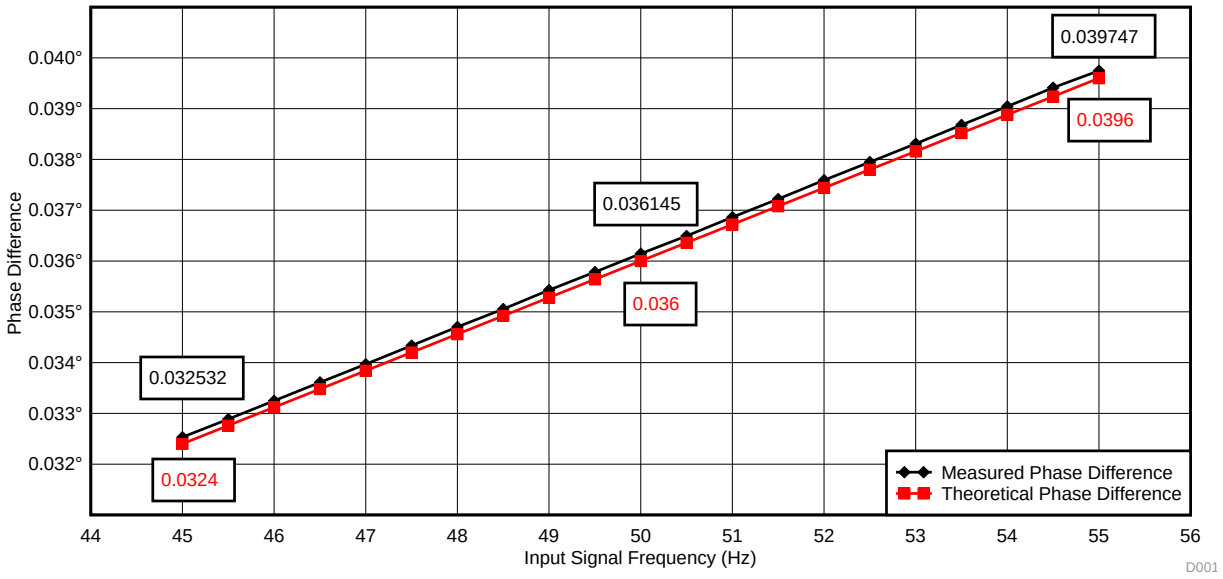


Figure 1. Measured and Theoretical Phase Difference Between Consecutive Channels

2 Theory of Operation

In modern power systems, accurate data collection, data processing, and fault diagnosis have become a primary focus of the current power grid. These factors are extremely critical to ensure correct operation of the power grid. The accurate measurement of electrical parameters of the various areas of the power grid helps to determine the operating status and running quality of the grid. Accurate measurement also helps to diagnose the failures or potential problems with the power network so that users can resolve such issues quickly without having any significant service impact. These parameters include the amplitude, frequency, and phase information, which is critical for performing harmonic analysis and calculating the power factor, power quality, and other parameters of the electrical power system.

The phase angle of the electrical signal on power network buses has always been a special interest to power system engineers. The active (real) power flow in a power line is very nearly proportional to the angle difference of the sinusoidal waveforms between voltages at the two terminals of the power line. Because many of the planning and operational considerations in a power network are directly concerned with the flow of real power, measuring phase angle differences (phase difference) across power transmission lines has been an industry concern for many years.

2.1 Basic Concepts of Phase and Phase Difference

Phase angle is one of the three important characteristics of an electrical signal (the other two are frequency and amplitude) that represent the AC signal in a transient state. In the context of periodic phenomena, such as a sinusoidal waveform, "phase angle" is synonymous with "phase". Phase angle is defined as the argument of the sine (or cosine) function as explained in Equation 1:

$$\mu(t) = A_m \sin(\omega t + \Phi) \quad (1)$$

where

- A_m is the amplitude of the waveform,
- ωt is the angular frequency of the waveform in radian/sec,
- Φ is the initial phase angle in degrees or radians.

If the positive slope of the sinusoidal waveform passes through the horizontal axis *before* t = 0, then the waveform has shifted to the left; so, Φ > 0 and the phase angle is positive. Likewise, if the positive slope of the sinusoidal waveform passes through the horizontal axis *after* t = 0, then the waveform has shifted to the right; so, Φ < 0 and the phase angle is negative. If the positive slope of the sinusoidal waveform passes through the horizontal axis *at* t = 0, then the waveform has no shift; so, Φ = 0 and the phase angle is zero. These concepts are explained below in Figure 2.

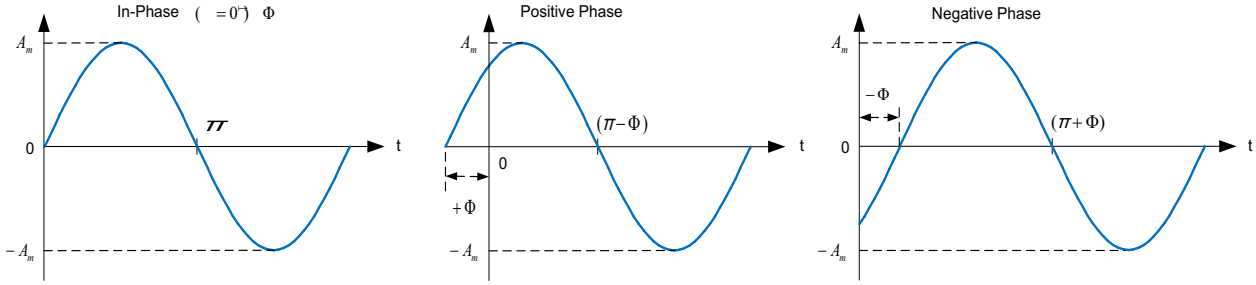


Figure 2. Illustration of Positive, Negative, and Zero Phase for Sinusoidal Signal

As an example, consider that two alternating signals such as voltage (V) and current (I) have the same frequency (f) in Hertz. If these signals have the same initial phase angle, then at any instant these voltage and current signals have the same phase. For any particular cycle, both these signals reach their positive and negative peaks (potentially different amplitudes) as well as pass the zero-crossover point at the same instant of time. Such signals are considered as “in-phase” signals, as shown in Figure 3:

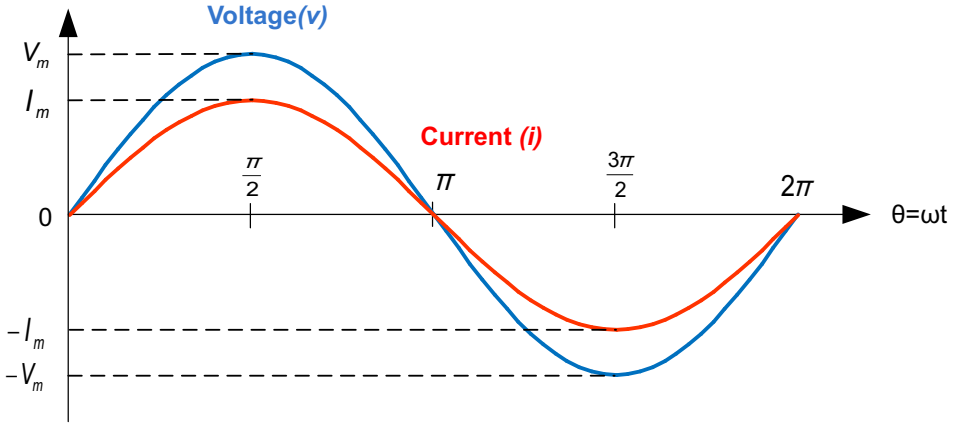


Figure 3. Two Sinusoidal Waveforms—"In-Phase"

Now consider that the voltage (V) and current (I) have a phase difference of 30° (so Φ = 30° or π / 6 radians). This phase difference (represented by Φ) remains constant at every instant of time because both these signals have the same frequency, (see Figure 4).

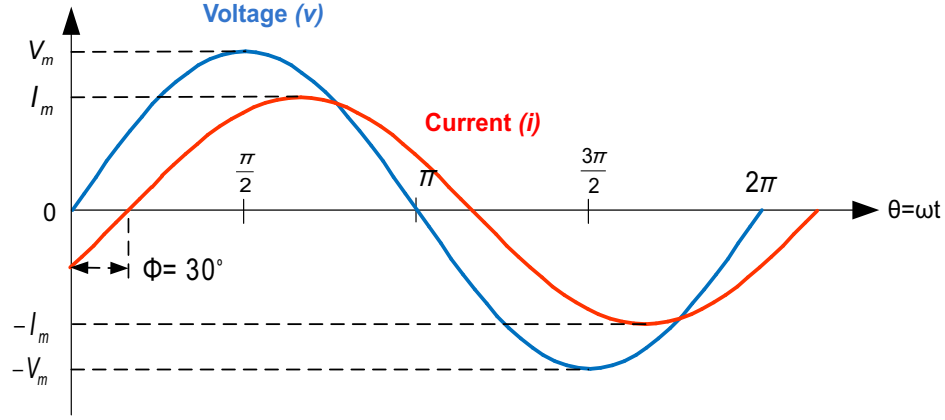


Figure 4. Phase Difference of Two Sinusoidal Waveforms

The voltage waveform in Figure 4 starts at zero along the horizontal reference axis. At that time instant, the current waveform is still negative in value and does not cross this reference axis until 30° later. The phase difference between the two waveforms results in the current signal reaching its peak values and crossing zero after the voltage waveform.

Two waveforms with a phase difference are considered "out-of-phase" by an amount determined by phi, Φ. In this example, Φ = 30°, so consider that the two waveforms are 30° out of phase. Consider also that the current waveform is "lagging" behind the voltage waveform by the phase angle, Φ, as explained in Equation 2:

$$V(t) = V_m \sin(\omega t),$$
$$I(t) = I_m \sin(\omega t - \Phi) \quad (2)$$

Likewise, if the initial phase angle of the current waveform has a positive value Φ and it crosses the reference axis before the voltage signal, then the current waveform is "leading" by a phase angle Φ, as explained in Equation 3.

$$V(t) = V_m \sin(\omega t),$$
$$I(t) = I_m \sin(\omega t + \Phi) \quad (3)$$

The relationship between any two waveforms and the resulting phase angle can be measured anywhere along the horizontal zero axis.

2.2 Phase Measurement Application

Phase measurement is referred to as the measurement of phase difference between two signals with the same frequency. With the development of science and technology, the application of phase measurement technology has penetrated into many areas and sectors, such as power electricity system, machinery sector, aerospace, geological exploration, seabed resource exploration, and so forth. Proper usage of phase measurement techniques can solve many problems associated with electrical and other non-electrical measurements. For example, when a power frequency voltage has been applied to an inductive load, the voltage and current signals have different phases. This phase difference introduces a phase shift in the AC signal when the AC voltage has been applied through a circuit or filter.

In electric power systems, phase measurement plays many important roles, some of which are as follows:

1. Through the measured phase difference (Φ) between current and voltage, the power factor can be obtained by the use of cos Φ. This method is useful for power quality calculations and several energy billing functions.
2. In a power system, measuring the phase difference between the voltage and current signals is often necessary. In many cases, this requirement is expanded to measure the phase in three-phase voltage or three-phase current systems. For example, in the main equipment transformer for power plants and transformer substations, measuring the phase relationship between the operating voltage and leakage current of the bushing is necessary. This measured phase difference can be used to calculate the dielectric loss of transformer bushing.
3. In a power system, when connecting two different power grids, accurately measuring the phase difference between the power frequency signals of these two power grids is necessary because the phase between these two electrical grids must be same.

In general, measuring the phase difference between electrical signals in a power system is a popular and necessary task. Some key applications of power automation include relay protection, measurement, and control system design. All these applications have similar requirements regarding measurement of the phase angles of voltage and current signals. These measurements are performed for each signal and subsequently calculate the phase difference between these signals.

2.3 Simultaneous and Non-simultaneous Sampling in Power Automation

Figure 5 shows a typical complete block diagram of the data acquisition system used in power automation application is shown in . Depending on the system and design requirements, the number of analog input channels is different. Also, depending on the selection of the ADC, the front-end amplifiers and external voltage reference circuits may vary. Many applications prefer to use ADCs with an integrated analog front end, an integrated reference for optimizing the available board space, or both. The power module and digital processing circuit, including the back-end isolation and interface, vary from one company to the other.

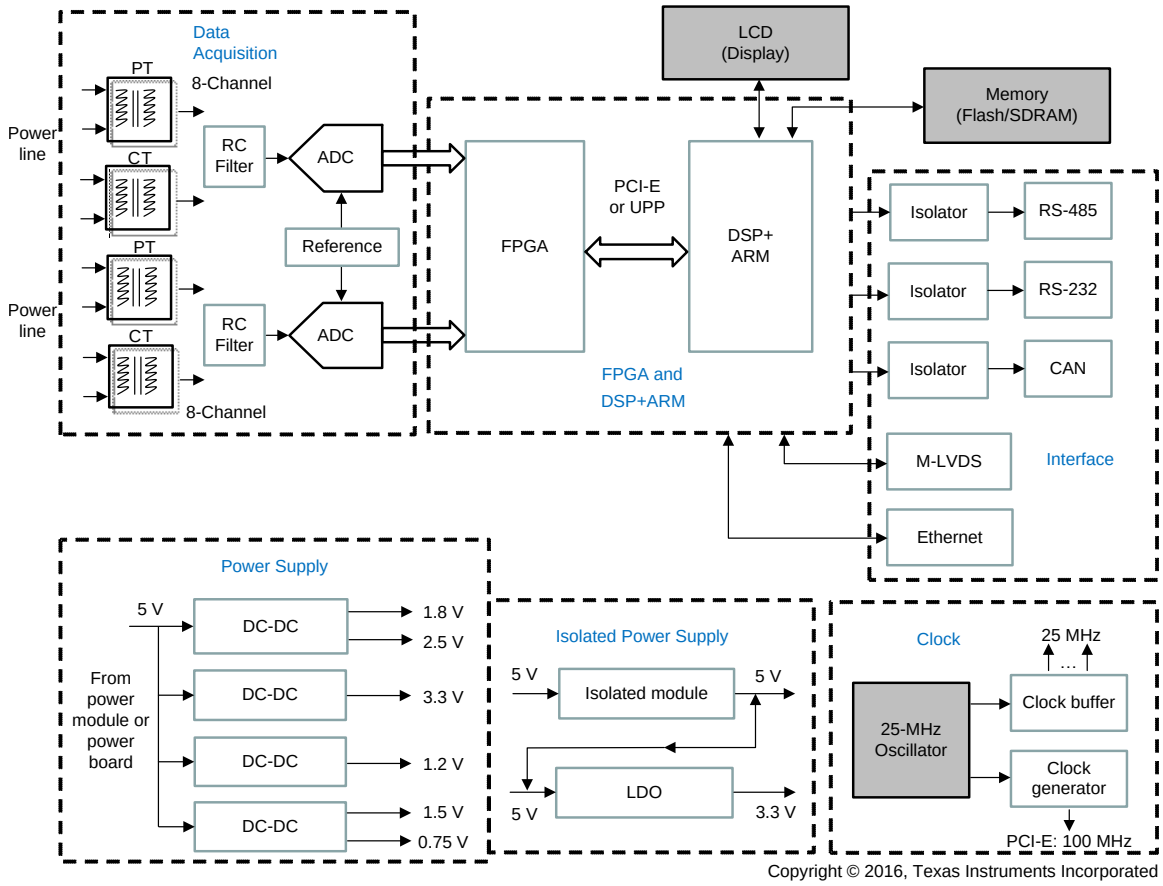


Figure 5. Typical Block Diagram for Data Acquisition System

In power measurement and relay protection systems, simultaneously sampling large numbers of voltage and current signals is often a requirement. Because the sampling instant for all channels can be synchronized, this method helps to maintain the phase information between the voltage and current channels. While this criteria is extremely important for high end and high voltage systems, this type of sampling may not be a critical requirement for medium and low voltage systems, which are typically deployed in substations of less than 35 kV. Such systems have a relatively relaxed accuracy requirement for the measurement of phase difference between the different input channels. Because of this relaxed accuracy requirement, a non-simultaneous sampling or multiplexed input ADC can be used for such applications. The additional phase difference caused as a result of sequential sampling in a multiplexed ADC is a constant factor of the sampling frequency and can easily be calculated and compensated by the resulting phase difference by implementing simple software.

To discuss simultaneous and non-simultaneous sampling, assume two signals with the same frequency (f) and the signal expressions are given in the following Equation 4 and Equation 5:

$$x(t) = A\sin(2\pi ft + \Phi_x) \tag{4}$$

$$y(t) = B\sin(2\pi ft + \Phi_y) \tag{5}$$

where

- A and B are the amplitudes of x(t) and y(t),
- Φ_x and Φ_y represent the phase of x(t) and y(t), respectively.

In an actual power grid system, the frequency of power signals is not fixed because of the impact of power grid fluctuation. The frequency of power signals vary because of the extensive use of nonlinear loads in a power grid system; so, generally the measured phase angles for x(t) and y(t) signals are expressed as the following Equation 6 and Equation 7:

$$\Phi_x = \Phi_{ix} + \Phi_{fx} \tag{6}$$

$$\Phi_y = \Phi_{iy} + \Phi_{fy} \tag{7}$$

where

- Φ_{ix} and Φ_{iy} are the useful initial phase angles,
- Φ_{fx} and Φ_{fy} are the phase angle errors because of signal frequency variation.

The phase difference between x(t) and y(t) signals can be calculated in Equation 8 as:

$$\Delta\Phi = \Phi_y - \Phi_x = (\Phi_{iy} - \Phi_{ix}) + (\Phi_{fy} - \Phi_{fx}) = \Delta\Phi_i + \Delta\Phi_f \tag{8}$$

where

- $\Delta\Phi_i$ is the initial phase difference,
- $\Delta\Phi_f$ is the phase error difference between different channels because of signal frequency variation.

2.3.1 Simultaneous Sampling

The primary advantage of a simultaneous sampling ADC in a power automation application is that it can sample many channels for the voltage and current signals at the same time and in such a way that there is no additional time delay between the sampled data across the different channels of a multi-channel system.

Figure 6 shows an example that samples input signals from four channels. These signals have the same frequency, which also means they have the same signal period, T. Let T_s be the sampling time interval, such that $T_s = 1 / f_{ADC}$ (f_{ADC} is the sampling frequency for ADC). The red dots on the curves indicate the positions of sampling points on each input signal.

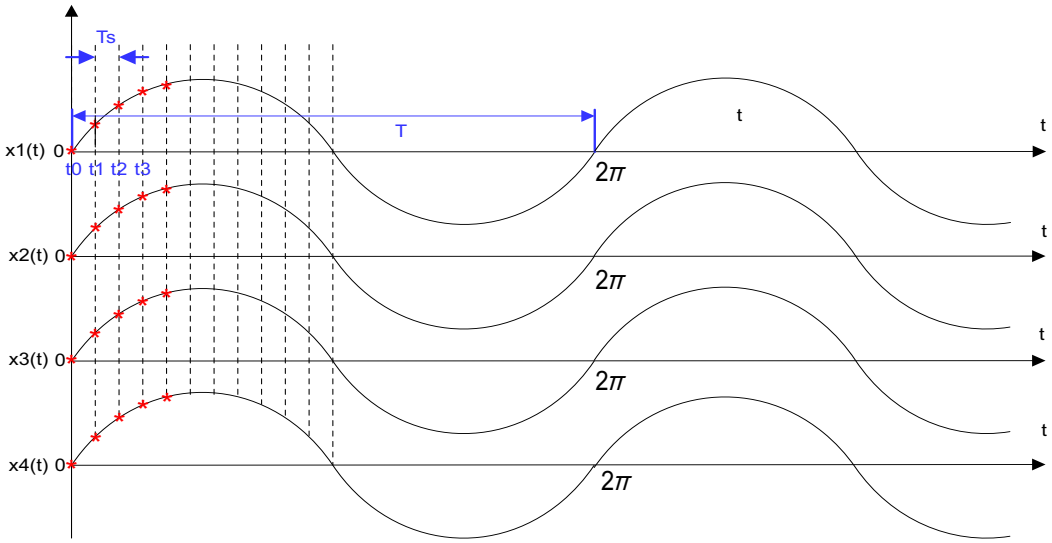


Figure 6. Simultaneous Sampling for Four Channel Signals

In the preceding diagram in Figure 6, all of the signals are sampled at the same sampling-time points t_0, t_1, ⋯ and t. No sampling time delay exists between the different channels because they are sampled at the exact same instant in time. According to Equation 6, Equation 7, and Equation 8, the phase difference between the different channels for this simultaneous sampling can be calculated in Equation 9

$$\Delta\Phi_{SS} = \Phi_y - \Phi_x = \Delta\Phi_i + \Delta\Phi_f \tag{9}$$

The phase error as a result of signal frequency variation can be minimized if the synchronous sampling (refer to Section 2.4.2.2) and signal frequency tracking methods (refer to Section 2.5.2.1 and Section 2.5.2.2) have been utilized. So if the phase angle error Φ_{fx} and Φ_{fy} is close to zero, the difference $\Delta\Phi_f = \Phi_{fy} - \Phi_{fx}$ is also close to zero and can be ignored. Equation 9 can be rewritten as the following Equation 10:

$$\Delta\Phi_{SS} = \Delta\Phi_i \tag{10}$$

As noted in Section 2.2, knowing the value for this initial phase difference between different signals is useful. This parameter is especially useful to system designers in power automation applications.

2.3.2 Non-Simultaneous Sampling

The name non-simultaneous sampling suggests that the ADC does not sample all input channels of a multi-channel system at the same exact time. This condition is very typical in the case of a multiplexed input ADC because the converter sequentially scans through the multiple input channels. Figure 7 shows an example of a four-channel system. If the total conversion cycle time of the ADC is T_s (the inverse of sampling frequency f_{ADC}, which is the sampling frequency for the ADC), then the additional time delay between two consecutive channels is $\Delta t = T_s$. These signals have the same frequency and the signal period is T, which is the inverse of the signal frequency, f_{in}.

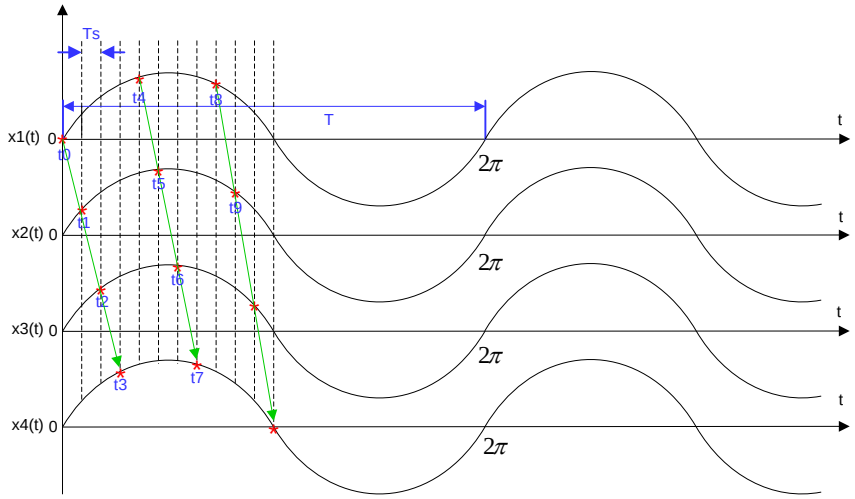


Figure 7. Non-Simultaneous Sampling for Four Channel Signals

To make the analysis simple and comparable with the simultaneous sampling analysis that Section 2.3.1 addresses, assume that the signals on these four channels have the same frequency. The ADC scans through these four channels sequentially starting with channel 1 at time t_0, followed by channel 2 at t_1, channel 3 at t_2, channel 4 at t_3, and then continues to loop around. At any particular instant in time, only one channel is sampled, which means the samples taken from the different channels correspond to different time domain information. The green arrow in Figure 7 explains this sequential sampling. As the sampling rate of the ADC increases, this additional time delay between any consecutive channels decreases and eventually the difference error becomes small enough that it can be ignored.

The phase difference between consecutive channels can be revised in the following Equation 11 considering the time delay caused by non-simultaneous sampling according to Equation 6, Equation 7, and Equation 8:

$$\Delta\Phi_{NSS} = \Phi_y - \Phi_x = \Delta\Phi_i + \Delta\Phi_f + \Delta\Phi_{mux} \tag{11}$$

where

- $\Delta\Phi_{mux}$ is the phase difference caused by non-simultaneous sampling structure and associated with the finite time delay.

As previously mentioned, when the synchronous sampling (refer to Section 2.4.2.2) and signal frequency tracking methods (refer to Section 2.5.2.1 and Section 2.5.2.2) have been utilized, the phase error as a result of signal frequency variation can be minimized. The Φ_{fx} and Φ_{fy} are close to zero and the difference $\Delta\Phi_f = \Phi_{fy} - \Phi_{fx}$ is also close to zero and can be ignored. Equation 11 can be rewritten as Equation 12:

$$\Delta\Phi_{NSS} = \Delta\Phi_i + \Delta\Phi_{mux} \tag{12}$$

If the initial phase angle of the x(t) and y(t) signals is zero or the same signal has been applied to these different channels of non-simultaneous sampling ADC, $\Delta\Phi_i$ is zero and $\Delta\Phi N_{SS} = \Delta\Phi_{mux}$.

According to the *Theory of Operation* introduction in Section 2 and Figure 7, the theoretical, additional phase difference between the two consecutive channels can be calculated in the following Equation 13:

$$\Delta\Phi_{mux} = \frac{T_s}{T} \times 360^\circ = \frac{f_{in}}{f_{ADC}} \times 360^\circ \tag{13}$$

where

- f_{in} is the signal frequency of input periodic signal (f_{in} = 1 / T),
- f_{ADC} is the sampling frequency for ADC as noted in the preceding (f_{ADC} = 1 / T_s),
- 360° (or 2π) is the phase for one cycle.

The ADS8688 is a 16-bit, 500-kSPS, eight-channel, non-simultaneous sampling ADC with an integrated analog front end and multiplexer. When the ADS8688 device operates on a maximum sampling frequency of 500 kSPS, the phase difference between the two consecutive channel signals can be calculated as Equation 14, but only if the input signal frequency is a 50-Hz sinusoidal waveform for all used channels and the initial phase angle is the same.

$$\Delta\Phi_{mux} = \frac{T_s}{T} \times 360^\circ = \frac{f_{in}}{f_{ADC}} \times 360^\circ = \frac{50}{500k} \times 360^\circ = 0.036^\circ \tag{14}$$

If all eight channels of the ADS8688 device have been selected and used, the maximum phase difference between the first (1st) and last (8th) channel is calculated in the following Equation 15:

$$\Delta\Phi_{mux-max} = 7 \times \Delta\Phi_{mux} = 0.252^\circ \tag{15}$$

A finite time delay exists between any two channels that the ADC converts, which is the only difference with simultaneous sampling. This delay is equal to one complete conversion cycle of the ADC, and for this reason, the delay can be minimized by operating the ADC at the maximum throughput. Even though such multiplexed systems introduce an additional phase difference between the input channels, the value of this phase difference is deterministic and small if the ADS8688 device operates at a fast sampling rate. This small difference makes it suitable and ideal to use this multiplexed ADC in relay protection and measurement systems including a power distribution network at a power automation market.

2.3.3 Non-Simultaneous Sampling With Phase Compensation

In most applications, system designers like to know the initial phase difference ($\Delta\Phi_i$) between any two signals of interest, which is the most vital information to the system. The preference is to minimize the phase difference error ($\Delta\Phi_f$) caused by the signal frequency variation. Achieve this task using the method introduced in Section 2.5, Appendix A, or by using other methods. As previously noted, these techniques can help to reduce this phase error by either tracking small changes in signal frequency or by adding the window function for the sampled sequence. The phase difference ($\Delta\Phi_{mux}$) caused as a result of a multiplexed input ADC is constant when the sampling rate of the ADC is fixed, which means that this phase difference can be subtracted directly from the measured total phase difference ($\Delta\Phi_{NSS}$) in software according to the theoretical value calculated by Equation 14. To summarize, use Equation 12, Equation 13, and Equation 14 to easily perform the phase calibration and compensation for every channel by software, as Section 2.5.1 details. This type of phase compensation technique makes a multiplexed ADC similar in functionality to a simultaneous sampling ADC in terms of the phase measurement requirements of a typical power automation application.

2.4 Principle of Phase Measurement

Several techniques of phase detection and measurement have been developed in the past few years that utilize the advancements in process and computer technology. Some detection methods are based on hardware circuitry and others are based on a combination of data sampling techniques and software calculation. Other popular techniques exist that utilize spectral analysis in the frequency domain or correlation analysis in the time domain.

Phase measurement methods based on pure hardware circuitry have several associated limitations. In such a method, the phase difference between two signals is first converted into a time difference. The sinusoidal signals under consideration go through signal conditioning and are converted into square wave signals using zero-crossover detection circuitry. The phase information of the signal can then be derived by pulse-width calculation of the square waveform signal; however, the accuracy of this technique is limited by the accuracy of the electronic components used in the design. Because of variations in the accuracy and performance of the electronic components (time resolution and comparator performance), achieving the same measurement accuracy between two channels is impossible and leads to calculation errors. The interference between the circuit components also leads to errors in the system. Furthermore, this technique is not very cost-efficient; hence, the scope of this document is limited to discussing the methods of phase detection and measurement based on a combination of data collection, software calculation, and processing techniques.

Many methods of phase measurement are viable and the following list provides several examples:

1. Phase measurement based on first-order linear interpolation
2. Phase measurement based on correlation analysis method
3. Phase measurement based on discrete Fourier transform (DFT)

Among the methods listed above, the actual software implementation of this TIPD167 design utilizes the DFT-based phase measurement technique. However, for the sake of clarity, Section A.1 details the other two non-Fourier transform methods.

2.4.1 Picket Fence Effect and Spectral Leakage for Fourier Transform

The DFT or fast Fourier transform (FFT) are the most widely-used measurement algorithms for measuring the phasor properties of a signal (frequency, amplitude, and phase angle) because both these methods offer good filtering options. However, two factors affect the accuracy of the measurement results:

- Picket fence effect – An effect caused by frequency domain sampling
- Spectral leakage effect – An effect caused by truncation of the signal

Picket fence effect:

The DFT or FFT algorithm produces a discrete frequency spectrum that calculates the power of an electrical signal at discrete frequencies. However, the true spectrum of the signal being analyzed may have peaks at frequencies between the lines of the DFT or FFT spectrum. In such a case, the peaks calculated in the DFT or FFT algorithm are at the exact frequencies. The term “picket fence” is derived from viewing an FFT spectrum, which is something like looking at a mountain range through a picket fence because of this effect. If the fundamental frequency peak of a signal falls exactly in the frequency bin calculated by the DFT or FFT algorithm, then the user can view the peak with the correct magnitude. In any other situation, the energy content of the signal at the fundamental frequency is shared by the adjacent DFT or FFT frequency bins, which may smear the peak or even make it undetectable.

One way to reduce the picket fence effect is to change the number of points in a time record by adding zeros at the end of the original record. Because of the increase in sample size, the size of the frequency bins in the DFT or FFT response is reduced; for this reason, the fundamental frequency peak can be adjusted to the appropriate frequency bin where the peak magnitude can be observed correctly.

Spectral leakage:

Spectral leakage results from an assumption in the DFT or FFT algorithm that the time record repeats exactly throughout all time. Acknowledging this assumption, signals in a time record are periodic at intervals that correspond to the length of the time record. When a user implements the DFT or FFT to measure the frequency content of data, the transforms assume that the finite data set is one period of a periodic signal. Therefore, the finite nature of the sampling record results in a truncated waveform with different spectral characteristics from the original continuous-time signal and the finite nature can

introduce sharp transition changes into the measured data. The sharp transitions are discontinuities. The discontinuities produce leakage of spectral information. Spectral leakage produces a discrete-time spectrum that appears as a smeared version of the original continuous-time spectrum and distorts the measurement in such a way that energy from a given frequency component is spread over adjacent frequency lines or bins.

2.4.2 Asynchronous and Synchronous Sampling

The user can choose from two solutions to minimize the effects of spectral leakage: the asynchoronous sampling method and synchronous sampling method.

2.4.2.1 Asynchronous Sampling Method

In some cases, an unknown signal is being measured or the integer-multiple relationship between an input and sampling control signal does not exist. In such a case, the user cannot guarantee that they are sampling an integral number of cycles and spectral leakage occurs; however, by performing the windowed DFT method over a non-integral number of sampling cycles, the user can minimize the effects of spectral leakage. The DFT method is a commonly used method for asynchronous sampling.

This asynchronous sampling is also called fixed time sampling or fixed frequency sampling. In this sampling method, the sampling interval or sampling frequency is fixed. This method does not require additional hardware and does not require adjusting the size of the data window N to track the signal frequency. The method only requires improving the accuracy through the use of software. The more commonly used method is the phase difference correction method. This method allows for the selection of the appropriate window function and phase correction method to achieve more accurate results. This solution uses a simple algorithm and has high accuracy, low implementation costs, and other advantages.

More information on asynchronous sampling methods, including a method to determine the phase difference between harmonics, can be found in Section A.2.

2.4.2.2 Synchronous Sampling Method

The most popular method to alleviate the effect of spectral leakage is to perform synchronous sampling of the input signals. This method of sampling enables the user to take the integral number of cycles of the measured signal, which can prevent or reduce spectral leakage.

Synchronous sampling is achieved by sampling the input signal at a sampling rate related directly to the signal frequency. Conceptually, this action makes the signal stationary for the purposes of the DFT or FFT analysis, which can then be applied to extract useful information about the frequency content of the signal.

Note that synchronous sampling is different from simultaneous sampling, which the following subsections address in further detail. Synchronous sampling is a mathematical technique used for analyzing the frequency content of any signal, whereas simultaneous sampling refers to taking time-domain samples in a multi-channel system.

To understand the fundamental principle behind synchronous sampling, consider a signal of frequency f_{IN} (cycle time, T_{IN}). Let the sampling frequency be f_S and the sampling time T_s, such that $f_s = 1 / T_s$. If N is the total number of samples taken and t_0, t_1 , $\cdots$ $t_{(N-1)}$, t_N are the sampling time instants, then the following Equation 16 and Equation 17 must hold true:

$$t_i - t_{i-1} = T_s \left(i = 0,1,2 \ldots N\right) \tag{16}$$

$$t_N - t_0 = N_C \times T_{IN} \tag{17}$$

where

- N_C is the integral number of cycles.

The following two conditions must be met to fulfill the basic criteria for synchronous sampling:

- An integral relationship must exist between the cycle time of the input signal and the total sampling time
- The time interval between two sampling points must be strictly consistent

The easiest way to achieve synchronous sampling is to take $T_s = T / N_C$ as the sampling interval. Under ideal conditions, this method can satisfy these two conditions of synchronous sampling.

The problem arises when the frequency of sampled signal changes during every cycle. Under such conditions, maintaining the sampling frequency as an integer multiple of the signal frequency is difficult and results in asynchronous sampling. If the DFT algorithm has been executed on data collected by asynchronous sampling, the DFT result is erroneous because of spectral leakage. Implementing preventive measures to avoid such errors is important in systems that require high precision measurements.

Having a deviation between the frequencies of the power grid signal compared to the standard power frequency is very common in power automation applications. If the sampling frequency has been kept constant, this frequency fluctuation around the rated frequency results in asynchronous sampling and spectral leakage. In particular, the calculated phase difference is very large, which does not meet the accuracy requirements of these applications.

This property establishes the importance of performing synchronous sampling for the purpose of measuring power grid signals. The process usually involves the following two steps:

1. Frequency measurement: This process involves measuring the exact input signal frequency using software or hardware techniques as detailed in Section A.3.
2. Frequency tracking: This process involves real-time tracking of the input signal frequency variation, which can also be done by hardware or software. In hardware implementation, the sampling frequency can be adjusted in a real-time manner. However, in software implementation, the sampling frequency can be kept constant, but some data processing can be done to achieve synchronous sampling.

Software synchronization method

The software synchronization can be implemented in the following two ways:

1. The first method involves measuring the signal frequency and keeping the data window size of the time domain samples constant, then adjusting the sampling interval based on the frequency calculated by the software algorithm and modifying the sampling frequency adaptively. In other words, this method involves adjusting the sampling interval in real time to meet the synchronous sampling condition. This method is suitable for a slow-changing frequency signal and also introduces an error when the software controls the hardware to implement the change of sampling interval time.
2. The second method involves measuring the signal frequency but keeping the sampling time (frequency) constant while the number of samples is changed to ensure that an integral number of cycles is captured in the window size of the time domain samples. The obvious problem with this technique is that to achieve perfect sampling, the value of N would have to be a non-integer. The value of N is rounded off to the nearest integer. The value of N is recalculated based on the frequency estimated from the previous iteration recursively. This method greatly improves the accuracy of phase measurement and compensation.

2.5 Principle of Software Algorithm

The software implementation of this TIPD167 design utilizes the software synchronization technique (second method in the preceding software synchronization methods) to calculate the frequency and phase angle of the signals using the DFT method.

This software synchronization algorithm measures the actual signal frequency based on DFT and does not require any additional hardware. The software synchronization algorithm then tracks the actual signal frequency but does not change the sampling frequency. The algorithm only adjusts the size of the data window based on the frequency estimated to achieve synchronous sampling, which means the algorithm can decrease the influence of frequency spectral leakage and improve the detection accuracy of frequency and phase of the power system signal. Additionally, Fourier transform only uses two cycles of data, so this algorithm does not require much calculation and is not time consuming. This algorithm is very suitable for real-time monitoring and measuring the phasor parameter in the replay protection application of a power automation market.

2.5.1 Phase Compensation Software Algorithm

The algorithm implemented in this design utilizes the DFT method to calculate and track the signal frequency, obtain the exact phase angle of the individual signal, and then calculate the phase difference and implement phase compensation. The entire algorithm consists of four steps:

1. Calculate the theoretical phase difference ($\Delta\Phi_{mux}$) introduced by the ADC as a result of multiplexing channels. Refer to Section 2.3.2 for a more detailed explanation of the phase difference introduced by ADCs as a result of multiplexing channels.
2. Estimate the frequency of the power system signal using the DFT method (frequency tracking).
   - Assume that the signal frequency of the power system is 50 Hz for the following example. The frequency of the system must be calculated before measuring the phase of the signal because the actual system frequency can be different based on the load conditions and other conditions.
   - Calculate the number of data points (N) for a cycle and collect two cycles of data from the ADC for the DFT calculation. Perform DFT on one cycle data at a time and calculate the phase of each cycle.
   - When the acquired signal frequency is exactly equal to the assumed 50 Hz, the phase angle of each cycle will be the same. When the signal frequency (red signal) has deviated from the expected 50 Hz (green signal) as shown in Figure 8, the phase angle of the second cycle will be different from the phase angle of the first cycle, which is directly related to the signal frequency.
   - Calculate the frequency of the signal using an equation based on the phase angles of two consecutive cycles.

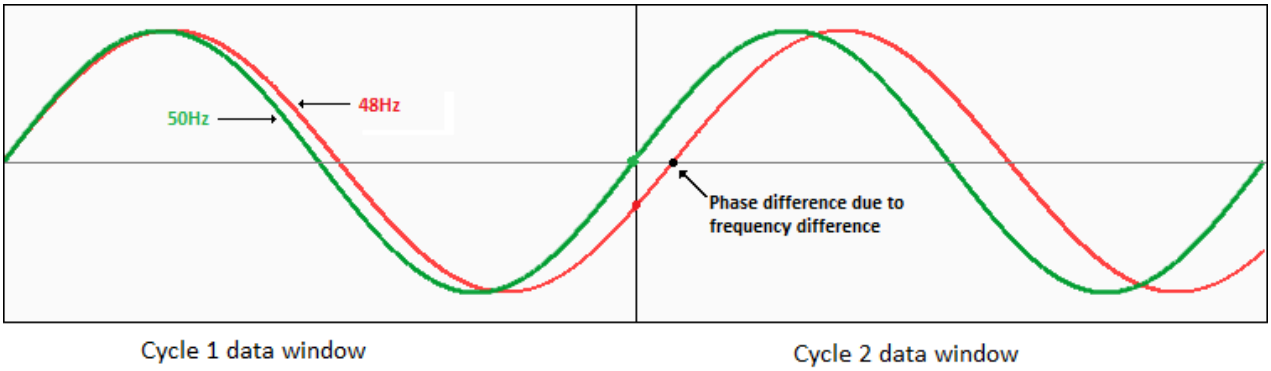


Figure 8. Phase Difference as Result of Frequency

3. Calculate the phase angle of all the signals in the system based on the estimated frequency.
   - After calculating the actual frequency of the power system signals, the user can calculate the phase angle of all the signals in the system.
   - Calculate the number of data points (N_1) based on the estimated frequency of the system, $N_1 = f_s / f_1$ (where, f_1 is the calculated signal frequency in the preceding *Step 2* and f_s is the sampling frequency per channel).
   - Using the data collected in *Step 2*, obtain one cycle of data for all the channels based on the preceding *Step 2*. Repeat the following process for each channel.
   - Perform the DFT method again on the data of cycle 1 and calculate the initial phase angle (Φ_i) of the signal where i is the channel index.
4. Compensate the phase difference for all the channels using the theoretical value calculated in *Step 1*.
   - Now compensate the phase difference introduced by the ADC multiplexing for all the channels.
   - The phase angle calculated in the preceding *Step 3* includes $\Delta\Phi_{mux}$ as a result of the multiplexing of channels by the non-simultaneous sampling ADC.
   - The phase difference $\Delta\Phi_{mux}$ is constant for the given ADC sampling rate and the signal frequency. This difference can be compensated for all the channels. Calculate the actual phase of channels $\Phi_{iCOMP} = \Phi_i - i \times \Delta\Phi_{mux}$.
   - Calculate the phase difference of channels with the reference channel $\Phi_{iDIFF} = \Phi_{REF} - \Phi_{iCOMP}$.

Figure 9 shows the software flow chart for these steps, f_{ADC} is the sampling frequency for ADC, f_s is the sampling frequency per channel, f_0 is the expected signal frequency ($T_0 = 1 / f_0$), Φ_1 is the phase angle of the first cycle, and Φ_2 is the phase angle of the second cycle.

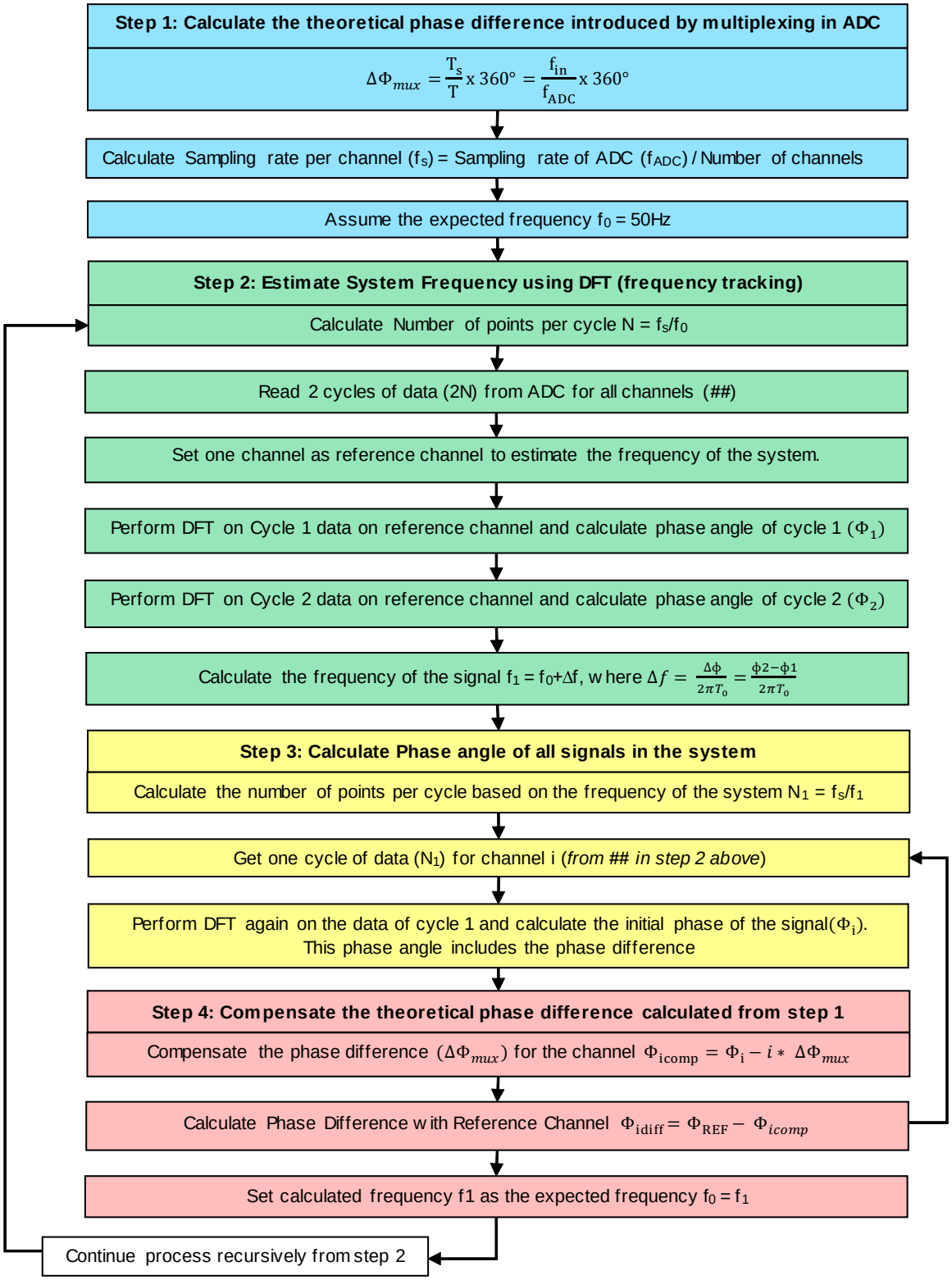


Figure 9. Phase Compensation Algorithm – Flow Chart

2.5.2 Principle of Phase Measurement With Software Synchronization

2.5.2.1 Frequency Measurement

The frequency measurement used in this design is a method of DFT-based frequency measurement based on the article *New Measurement Technique for Tracking Voltage Phasors, System Frequency and Rate of Change of Frequency* [6].

Assuming the input signal for the power system is the voltage signal without any harmonics:

$$u(t) = u_m \sin(\omega t + \Phi_0) \tag{18}$$

where

- f_0 is the rated fundamental frequency of the power system (basically 50 Hz),
- Φ_0 is the initial phase angle.

The actual system frequency is f, $f = f_0 + \Delta f$, so:

$$u(t) = u_m \sin(2\pi f_0 t + 2\pi \Delta f t + \Phi_0) \tag{19}$$

The following Equation 20 can be derived if using $\Phi(t) = 2\pi\Delta f t + \Phi_0$:

$$\frac{d\Phi}{dt} = 2\pi\Delta f \tag{20}$$

The phase angle difference between two consecutive cycles is:

$$\Delta\Phi = \Phi_2 - \Phi_1 \tag{21}$$

where

- Φ_1 is the phase angle calculated based on the data in the first cycle,
- Φ_2 is the calculated phase angle based on the data in the second cycle (which follows first cycle closely).

The frequency deviation Δf between the actual frequency f and power frequency f_0 can be obtained according to Equation 20 and is shown in Equation 22:

$$\Delta f = \frac{1}{2\pi}\frac{d\Phi}{dt} = \frac{1}{2\pi}\frac{\Phi_2 - \Phi_1}{N\Delta t} = \frac{1}{2\pi}\frac{\Phi_2 - \Phi_1}{N\frac{T_0}{N}} = \frac{1}{2\pi}\frac{\Phi_2 - \Phi_1}{T_0} = \frac{\Delta\Phi}{2\pi T_0} \tag{22}$$

where

- N is the sample number per cycle in $\Delta t = \frac{T_0}{N}, T_0 = \frac{1}{f_0}$

Equation 23 calculates the actual measurement frequency:

$$f = f_0 + \Delta f = f_0 + \frac{\Delta\Phi}{2\pi T_0} \tag{23}$$

To measure Φ_1 and Φ_2 and obtain $\Delta\Phi$, the following DFT equations in Equation 24 can be used to calculate the phase of sampled data of u(t).

$$U_R = \frac{2}{N}\sum_{k=0}^{N-1} u_k \cos\frac{2\pi}{N}k$$
$$U_I = \frac{2}{N}\sum_{k=0}^{N-1} u_k \sin\frac{2\pi}{N}k \quad (24)$$

So:

$$\tan\Phi = \frac{U_I}{U_R}$$
$$\Phi = \arctan\frac{U_I}{U_R} \quad (25)$$

However, when the frequency deviation between the actual signal frequency and rated power frequency is large, the calculated phase based on single cycle data is not accurate because of spectral leakage and the fence effect. When the deviation is large, the points collected for one cycle based on ideal frequency do not represent a complete cycle of data. The DFT results based on an incomplete cycle data have an erroneous phase angle value. This error in phase angle leads to the error in estimating the frequency of the power system signal based on Equation 23, so the DFT algorithm requires a complete cycle of data for synchronous sampling to obtain accurate results. The frequency tracking method can be employed to calculate the signal frequency accurately.

2.5.2.2 Frequency Tracking

The real number of samples per cycle should be $N' = f_s / (\Delta f + f_0)$ to avoid generating spectral leakage when implementing the Fourier transform as the Equation 24 for a data acquisition system with constant sampling rate (f_s).

The Δf is the deviation between the true signal frequency and rated power frequency and N' can only be an integer, which leads to an error. Equation 26 shows the number error of sampling points for every period of sinusoidal wave:

$$\Delta N = N - N' = \frac{f_s}{f_0} - \frac{f_s}{\Delta f + f_0} = \frac{f_s \Delta f}{f_0(\Delta f + f_0)} \quad (26)$$

When Δf is large, the deviation of real and imaginary parts when calculating from Equation 24 is great; therefore, the corresponding phase deviation is also large and the accuracy of the frequency obtained by the phase difference $\Delta\Phi$ is affected. A general approach to reducing the error resulting from the number of samples per cycle is to use a PLL to adaptively adjust the sampling frequency and achieve the sampling synchronization between the sampling period and the actual cycle. However, adjusting the sampling period requires additional hardware, which increases the system error and cost. As a result, a software frequency tracking method is employed by changing the number of sample points per cycle for DFT.

Equation 26 shows that a larger Δf results in a larger ΔN and a greater phase difference, too, which means that the user can utilize the software to track the frequency and reduce ΔN by reducing Δf.

For the data obtained by sampling with the constant sampling frequency per channel (f_S), firstly $N = f_s / f_0$ can be calculated (f_0 is the power frequency and typically is 50 Hz) and used into Equation 24. At this time, the deviation of the sampling points is ΔN, then the preliminary estimated frequency (f_1) can be calculated according to Equation 21, Equation 22, Equation 23, and Equation 24. Then f_1 is used to change the size of the data window, $N_1 = f_s / f_1$ (N_1 may be a non-integer to be chosen according to the rounding method, it can be implemented in software as $N\_1 = \text{int}(f_s / f_1 + 0.5)$, at this time the fundamental frequency of DFT has become $f_0' = f_s / N_1$, the new deviation of the sampling points ($\Delta N'$) is significantly smaller than ΔN.

Frequency tracking is the method of using the calculated frequency from the previous iteration in the next iteration as the input to calculate the number of data points. This method can accurately track the signal frequency by continuous iteration.

2.5.2.3 Phase Angle Calculation

Perform DFT and calculate the phase angle of the signal. The data points are sampled synchronously based on the estimated frequency (as explained in the preceding subsections) to improve the accuracy of the phase angle measurement. The following Equation 27 and Equation 28 can be obtained using the value for N_1, which is calculated by the frequency tracking in Equation 24 and Equation 25:

$$U'_R = \frac{2}{N_1}\sum_{k=0}^{N_1-1} u_k \cos\frac{2\pi}{N_1}k$$
$$U'_I = \frac{2}{N_1}\sum_{k=0}^{N_1-1} u_k \sin\frac{2\pi}{N_1}k \quad (27)$$

$$\Phi' = \arctan\frac{U'_I}{U'_R} \quad (28)$$

Using this method, calculate the phase angle for every channel in the system. For the subsequent iterations, calculate N (number of data points) based on the frequency from the previous iteration. When performing the phase angle calculation with a new value for N, the phase error as a result of signal frequency variation (see Section 2.3) can be minimized by using this signal frequency tracking method.

With the continuous iterative manner, the phase angle is continuously being updated. After calculating the phase angle per channel by this method, the user can identify the phase difference between the channels of the ADC.

3 Component Selection

This section explains the selection of the various components specific to this TI Design. This topic mainly addresses the selection of a high precision ADC with integrated analog front-end circuit and components of an external RC filter circuit for interfacing with the voltage and current outputs from transformers in a power automation application.

3.1 ADC Selection

Consider the following key ADC specifications to design a high-performance data acquisition system for power automation and relay protection applications:

- SAR architecture is preferable because it allows data capture with minimum latency
- High resolution (≥ 16 bits)
- Multi-channel inputs are required because a typical application has multiple voltage or current lines, which must be monitored for power quality and relay protection
- Bipolar input ranges up to ±10 V are preferable
- High DC precision (low gain and offset errors) is required for overall system accuracy
- Low temperature drift is preferable because temperature calibration is difficult to perform
- Integrated voltage reference to cover more applications
- Single 5-V supply operation is preferable to avoid the generation of a ±15-V supply for a high-voltage analog front end
- High input impedance (>1 MΩ) for direct interface with the transformer outputs

The ADS8688 has been selected for this TI Precision Design because it meets the aforementioned requirements. The ADS8688 is a 16-bit, 500-kSPS, eight-channel ADC with an integrated analog front end, multiplexer, and precision reference. This device supports bipolar input ranges up to ±10.24 V with a single 5-V supply. The following list outlines the key features of this ADC that make it suitable for this application:

- 16-bit resolution at a 500-kSPS sampling rate
- Selectable and bipolar input ranges up to ±10.24 V
- Eight-channel multiplexor (MUX) with auto and manual channel scan
- Constant and high input impedance of 1 MΩ
- High performance:
  - 0.05% (maximum) for gain error, 1 ppm/°C (typical) and 4 ppm/°C (maximum) for drift
  - ±0.75 mV (maximum) offset error with 3-ppm/ºC maximum drift
- On-chip 4.096-V reference with low drift

3.2 Design Consideration for ADC Front-End Circuit

Most system engineers of multi-channel power line protection and measurement systems prefer to use an ADC with an integrated analog front end to avoid using external ADC drivers and only use passive RC filters or multi-stage filters to reduce the noise of the input signal. The use of proper resistor and capacitor values for the filter is important to avoid any system errors in terms of DC and AC performance.

Figure 10 is a typical front-end circuit design using the ADS8688 and a balanced low-pass RC filter, interfacing with the outputs of a current transformer (CT) or potential transformer (PT). This example shows only one channel for the sake of simplicity.

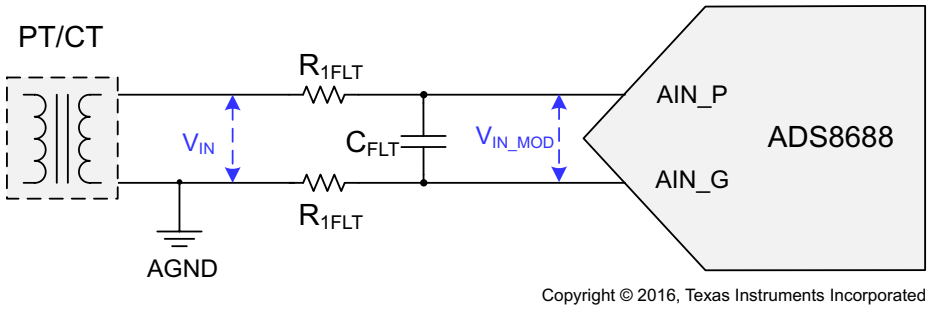


Figure 10. ADS8688 Front-End Circuit Design

A balanced RC filter at the input is required to reduce any impact on the offset error of the system. The operating frequency for CT isolation, PT isolation, and measurement transformers is 50 Hz. The amplitude of the input signal is usually ±10 V or ±5 V. Because the input signal frequency is very low, the cut-off frequency of the RC filter has been designed to be very low. This low input signal frequency also helps to significantly reduce the system noise. In a typical power automation application, the cut-off frequency of the input RC-filter is guided by the number of harmonics of the fundamental signal that must be measured by the system. Because the typical signal frequency is f_{typ} = 50 Hz or 60 Hz, if the number of harmonics to be measured is equal to N, then the RC-filter cut-off frequency is governed by the following Equation 29:

$$f_{-3dB} \geq N \times f_{typ} \tag{29}$$

For a low cut-off frequency filter, the values of the resistor (R_{FLT}) and capacitor (C_{FLT}) are typically very large. The large resistor also provides isolation between the SAR ADC and signal resource because the signal chain does not have a front end amplifier. However, the value of the resistor cannot be made very large because it has an adverse effect on the system DC performance of the whole signal chain. The external resistor R_{FLT} causes an additional gain error, which is a function of the input impedance of the ADC. A larger value for the ADC input impedance results in a smaller system gain error because of the external resistor and vice-versa.

Assuming this $R_{1FLT} = R_{2FLT} = R_{FLT}$:

$$V_{IN\_MOD} = V_{IN} \times \left(\frac{Z_{IN}}{R_{FLT} + Z_{IN}} \right) \tag{30}$$

$$Gain\_Error(\%) = \left(\frac{V_{IN} - V_{IN\_MOD}}{V_{IN}} \right) \times 100 = \left(\frac{1}{1 + \frac{Z_{IN}}{R_{FLT}}} \right) \times 100 \tag{31}$$

The previous assertions and calculations in Equation 30 and Equation 31 indicate that special consideration is required to design the front-end RC filter. This TI Precision Design uses the ADS8688 device, which integrates front-end amplifiers and provides a constant resistive input Impedance of 1 MΩ. This high input impedance helps to minimize the gain error and makes the ADS8688 a suitable device for applications that do not use a front-end amplifier.

As an example (using Equation 31), if choosing 1 kΩ for R_{FLT}, then the gain error is equal to 0.1%. The gain error increases to approximately 1% if the value of R_{FLT} has been increased to 10 kΩ.

The primary goal of this design is to showcase the importance of phase measurement and phase compensation in the case of a multiplexed ADC. For this reason, the cut-off frequency has been designed to be slightly higher to effectively measure a sufficient number of harmonics of the fundamental signal.

The selected value of R_{FLT} is 357 Ω and C_{FLT} is 10 nF, which gives a f_{3dB} of approximately 23 kHz.

4 TINA-TI™ Software Simulation for ADS8688

The TINA-TI™ SPICE model for the ADS8688 or 8684 ADC can be used to evaluate the performance of an entire signal chain. As an example, the TINA-TI schematic that Figure 11 shows uses the ADS8688 in MODE1 (±2.5 × VREF input range).

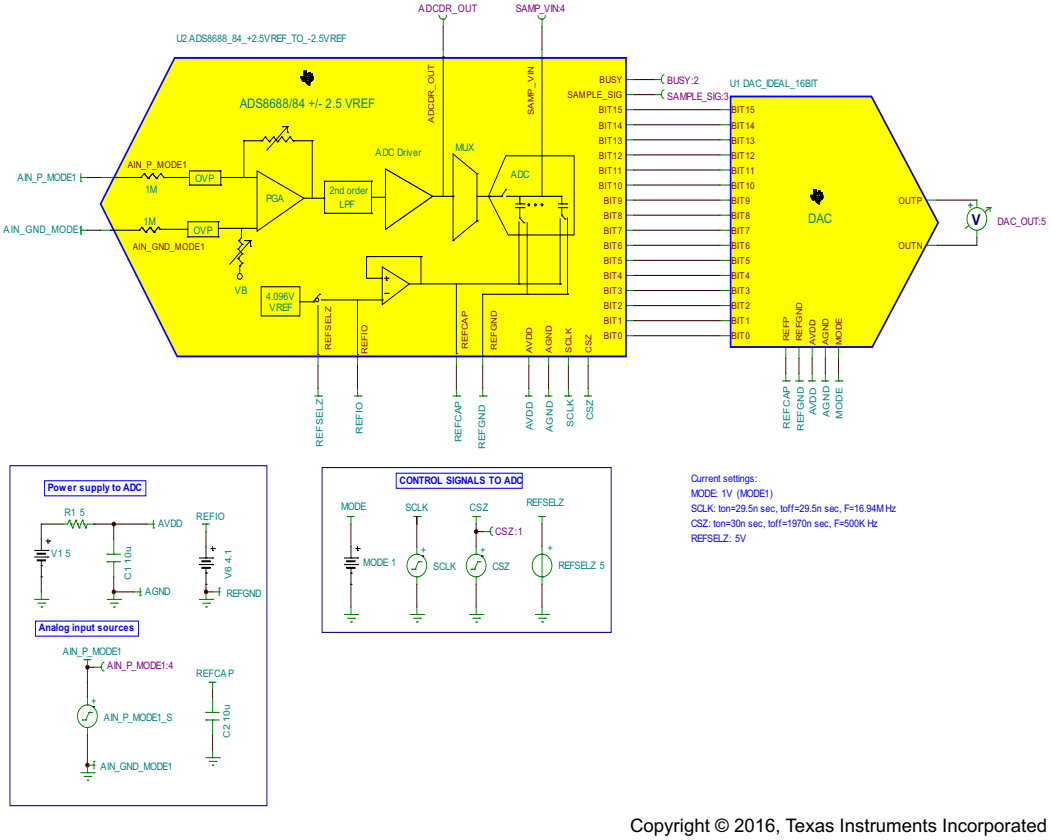


Figure 11. ADS8688 TINA-TI™ Schematic

The TINA-TI SPICE model for the ADS8688 is used to evaluate the performance of the entire signal chain. This SPICE model for the ADC accurately models the input structure of the ADC, including the input impedance, magnitude response, and phase response of the input low-pass filter. Each input range of the ADS8688 has been modeled as a separate model. This TI Precision Design utilizes MODE1, which represents the ±10.24-V input range of the ADC. The model also includes the dynamic loading of the REFP input pin. This inclusion helps to optimize the design of the external reference driver circuit.

This circuit in the preceding Figure 11 has been simulated to check the transient response of the ADC for a ±10.24-V input signal and the accurate settling of reference input voltage for each conversion. The following subsections provide simulation details and results.

4.1 Transient Simulation of Input Front-End Circuit

The TINA-TI schematic shown in Figure 11 is used to check the transient response at the inputs of the ADC during the sampling phase. Figure 12 shows the simulated time-domain response for the circuit. The transient plot on the top shows one cycle of a 1-kHz sinusoidal waveform with an amplitude of ±10.24 V applied at the input of the ADS8688 device. In MODE1, the ADC operates in the ±10.24-V input range and the sampling rate is 500 kSPS. The signal AIN_P_MODE1 represents the actual input signal at the input of the ADC and the signal SAMP_VIN shows the output of the input sample-and-hold circuit of the ADC. The lower plot shows the same waveform zoomed in on a time scale for more detail. The curves have been collated together to show the timing for conversion and sampling phases. Based on the simulated waveform, it is clear that the sampled signal tracks the input signal during sampling and stays on hold when the ADC is converting, which starts from the falling edge of the CSZ (chip select) signal.

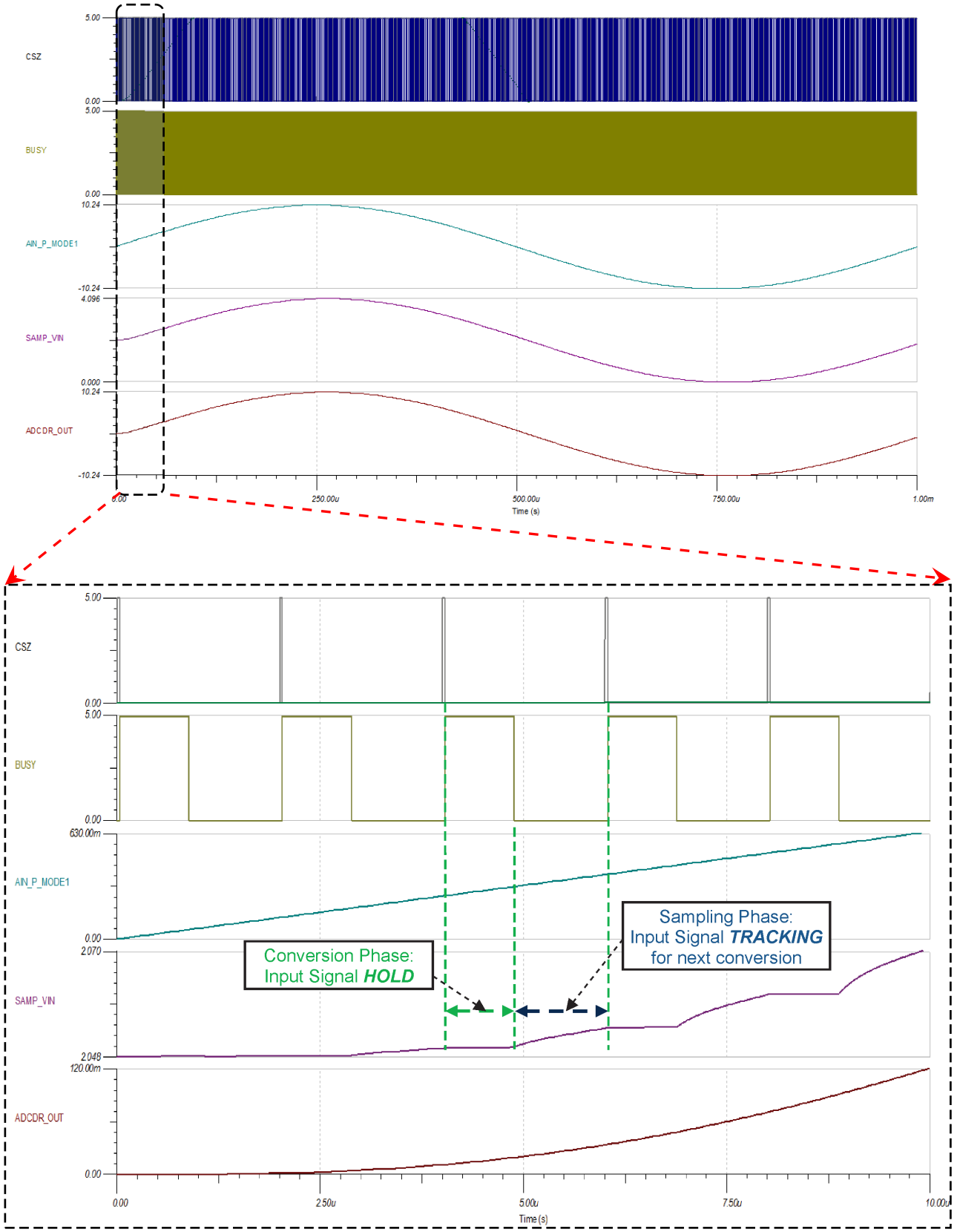


Figure 12. TINA-TI™ Model Simulation for Sinusoidal Waveform Input—MODE1

4.2 Transient Simulation of Internal Reference Buffer

The ADS8688 TINA-TI schematic shown in Figure 11 can also be used to check the settling of the internal reference buffer output, also known as the REFCAP pin of the ADS8688. The reference should settle to less than the LSB of the ADC to maintain the specified system performance. The size of the LSB for the ADS8688 using VREF = 4.096 V is equal to 62.5 µV. According to the transient simulation plot shown in Figure 13, with a 10-µF decoupling capacitor and internal reference enabled, the maximum voltage drop at the REFCAP pin during conversion time is ΔVREF = 3.61 µV, which is significantly less than the size of the LSB. This observation also validates that the internal reference voltage has settled to a sufficient accuracy to maintain the performance.

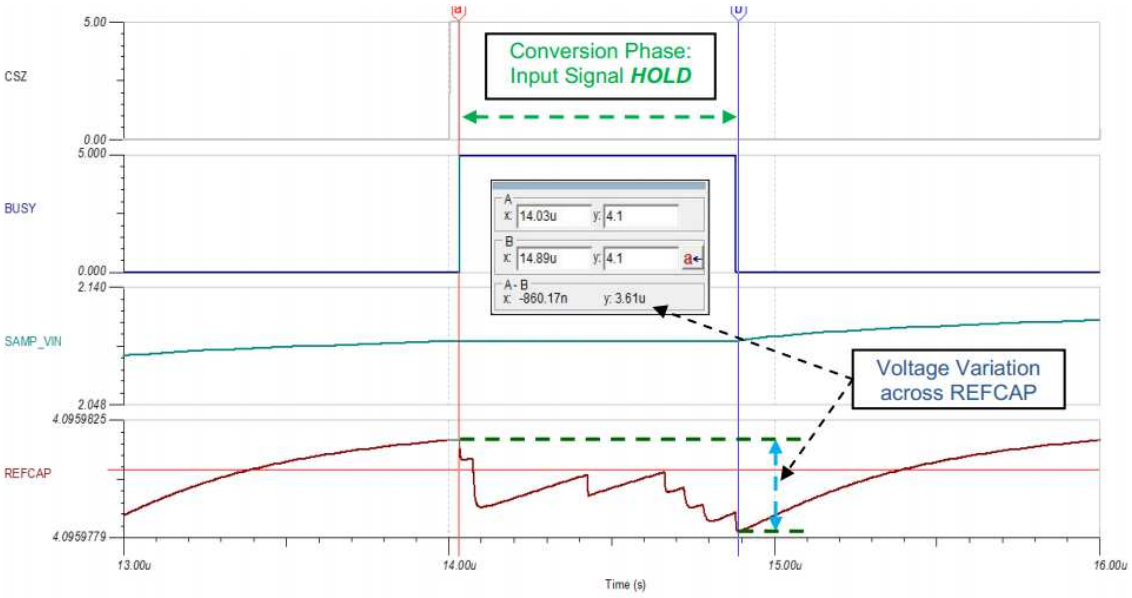


Figure 13. Voltage Change Across Reference Capacitor During Conversion Time

The minimum recommended value of the capacitor (C2) on the REFCAP pin is 10 µF. This capacitor is important to regulate the loading on the REFCAP pin. If the size of the capacitor has been reduced, the voltage change across the reference capacitor is different. This effect can be simulated and Table 2 shows the resulting voltage drop with typical capacitors.

Table 2. Voltage Change for Different Reference Capacitors

REFCAP CAPACITOR	VOLTAGE CHANGE ΔVREF	LESS THAN LSB
10 µF	3.61 µV	Yes
1 µF	93.77 µV	No
4.7 µF	7.43 µV	Yes
22 µF	1.68 µV	Yes

5 Verification and Measured Performance

In this section, the measurement tests and simulations demonstrate how the phase compensation software developed in this design can compensate the additional phase delay introduced by a multiplexed ADC, thus making the multiplexed ADC equivalent in performance to a simultaneous sampling ADC.

From Equation 9, Equation 10, and Equation 11, the only difference between non-simultaneous and simultaneous sampling is the phase difference ($\Delta\Phi_{mux}$) as a result of the additional phase delay, which is caused by the sequential switching of a multiplexed input ADC. As Section 2.3.2 addresses, $\Delta\Phi_{mux}$ is constant and small enough for a fast throughput ADC like the ADS8688 device, which operates at a maximum sampling rate of 500 kSPS. The constant theoretical value calculated in Equation 15 and Equation 16 can be verified by the phase compensation algorithm (see Section 2.5) and the compensation software, which is specially developed using the algorithm and integrated in the ADS8688EVM-PDK graphical user interface (GUI) software. This software is available for download on TI.com in the ADS8688EVM-PDK tool folder and can be used together with the ADS8688EVM to confirm the measurements in this precision design.

The verification of the phase compensation algorithm described in this design requires the experimental setup shown in Figure 14. This setup utilizes a signal generator (AP2722), which provides multiple analog inputs to the ADS8688 EVM (ADS8688EVM-PDK). The ADS8688EVM board interfaces with a computer through an interface module known as the simple capture card, which is a part of the ADS8688EVM-PDK package. The phase compensation software is a part of the evaluation software or ADS8688EVM GUI.

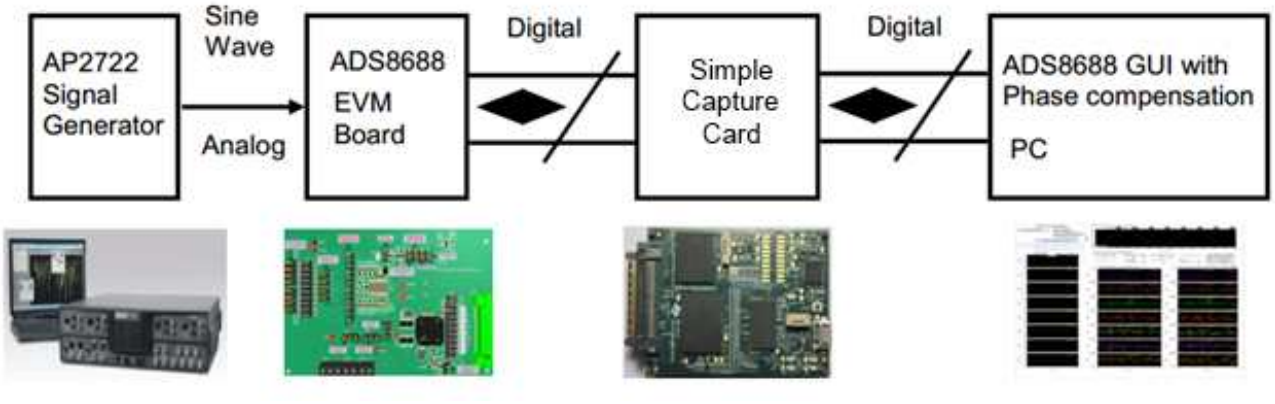


Figure 14. Test Connection and Flow Chart

According to the topics addressed in Section 2.3.2, when the same signal has been applied to all the input channels of an ADS8688 device, the initial phase difference ($\Delta\Phi_i$) is zero because the initial phase angle of every channel is the same. The phase error ($\Delta\Phi_f$) because of frequency change is close to zero because the signal frequency tracking algorithm is used in this software; so, the total phase difference ($\Delta\Phi_{NSS}$) is the only phase difference ($\Delta\Phi_{mux}$) because of the multiplexed input structure of the ADC.

The following subsections describe multiple test conditions for the verification, which are common in typical power automation applications.

5.1 Verification Analysis—Case 1: All Input Signals Same With 50-Hz Frequency

In this case, one pure sinusoidal waveform with a 50-Hz signal frequency and ±10-V signal amplitude from the AP2722 signal generator is applied to all the input channels of the ADS8688 device on the EVM board. The ADS8688EVM board is connected to the computer through the simple capture card. The software can be executed by using the menu option *Smart App → Phase Compensation* in the GUI software. The test connection for this case test is the same as the connection shown in Figure 14.

To set up this test, the sampling frequency has been set to 500 kSPS in this example and one of the analog input channels has been selected as the reference channel. In this example, channel *CH0* has been chosen as the reference channel. On the start of conversion, the phase difference between different channels can be obtained using this software. The tested phase difference value can be compared with the theoretical value according to the calculation in Equation 15 for two consecutive channels. The same

comparison can also be made for any two input channels with the worst-case condition being the farthest two channels. Figure 15 shows a screenshot of the software for this test condition. Note that the graphs of the phase difference with compensation and without compensation look similar; however, the scale of the vertical axis reflects that the phase difference between channels becomes very close to zero after the compensation algorithm.

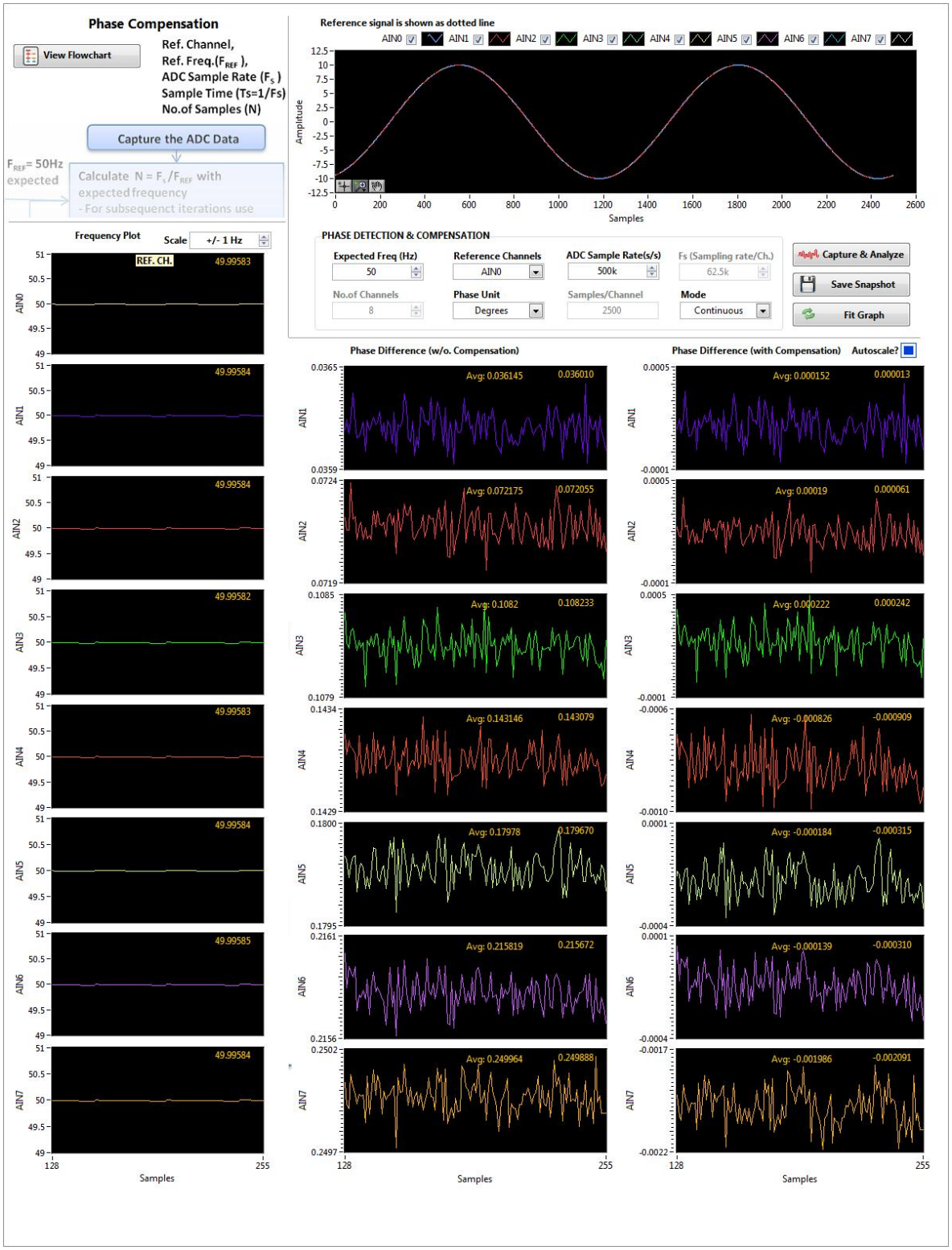


Figure 15. Screenshot of Verification Test 1

Further analysis of this test result indicates that the phase difference ($\Delta\Phi_{mux}$) between different channels is constant. For two consecutive channels (CH0 and CH1, CH1 and CH2, CH2 and CH3, and so on), the measured value is very close to the theoretical value of 0.036° for a 50-Hz input signal, sampled at 500 kSPS. With CH0 as the reference channel, the phase difference between CH0 and CH2 is approximately 0.072°, between CH0 and CH3 is approximately 0.108°, and the worst-case difference of CH0 and CH7 is close to 0.252°. This test demonstrates that the measured results match the theoretical calculation value.

Table 3 shows the test result summary for Figure 15.

Table 3. Measured Phase Difference Summary Before and After Compensation—Case 1(1)

50-Hz SINUSOIDAL INPUT	UNCOMPENSATED PHASE DIFFERENCE Δθ	COMPENSATED PHASE DIFFERENCE Δθ
Phase difference (Ch1 – Ch0)	0.036145º	0.000152º
Phase difference (Ch2 – Ch0)	0.072175º	0.00019º
Phase difference (Ch3 – Ch0)	0.1082º	0.000222º
Phase difference (Ch4 – Ch0)	0.143146º	–0.000826º
Phase difference (Ch5 – Ch0)	0.17978º	–0.000184º
Phase difference (Ch6 – Ch0)	0.215819º	–0.000139º
Phase difference (Ch7 – Ch0)	0.249964º	–0.001986º

(1) Channel 0 is the reference channel for the test.

5.2 Verification Analysis—Case 2: All Input Signals Same With Changing Frequencies

The phase compensation software has been designed to adopt the frequency tracking method. The software can quickly track any changes in the input signal frequency with only two cycles of sampled data for every input channel. In a typical power automation application, the signal frequency is not fixed and slowly fluctuates around a typical value of 50 Hz; however, the maximum frequency change does not exceed ±5 Hz (an approximate 45- to 55-Hz range). This change in input frequency can be tracked by the software, which can then eliminate the phase error (Φ_{fi}, phase error because of signal frequency variation) and precisely calculate the phase angle for every channel. This change in frequency is the same for all channels in a typical power system, so the phase difference between different channels is also constant and independent of the signal frequency change. However, the theoretical calculation error increases with the increase of deviation between the actual and standard signal frequency. The tested value of phase difference for different channels is the basically as the test in Section 5.1.

The test connection for this test is the same as shown in Figure 14. The sinusoidal input signals are still generated from the AP2722 signal generator but the frequency varies from the standard 50-Hz signal. The software screenshot for the test is shown in Figure 16 and the illustration uses an input signal frequency of 50.5 Hz as an example. Note that the graphs of phase difference with compensation and without compensation look similar, but the scale of the vertical axis reflects that the phase difference between channels becomes very close to zero after the compensation algorithm.

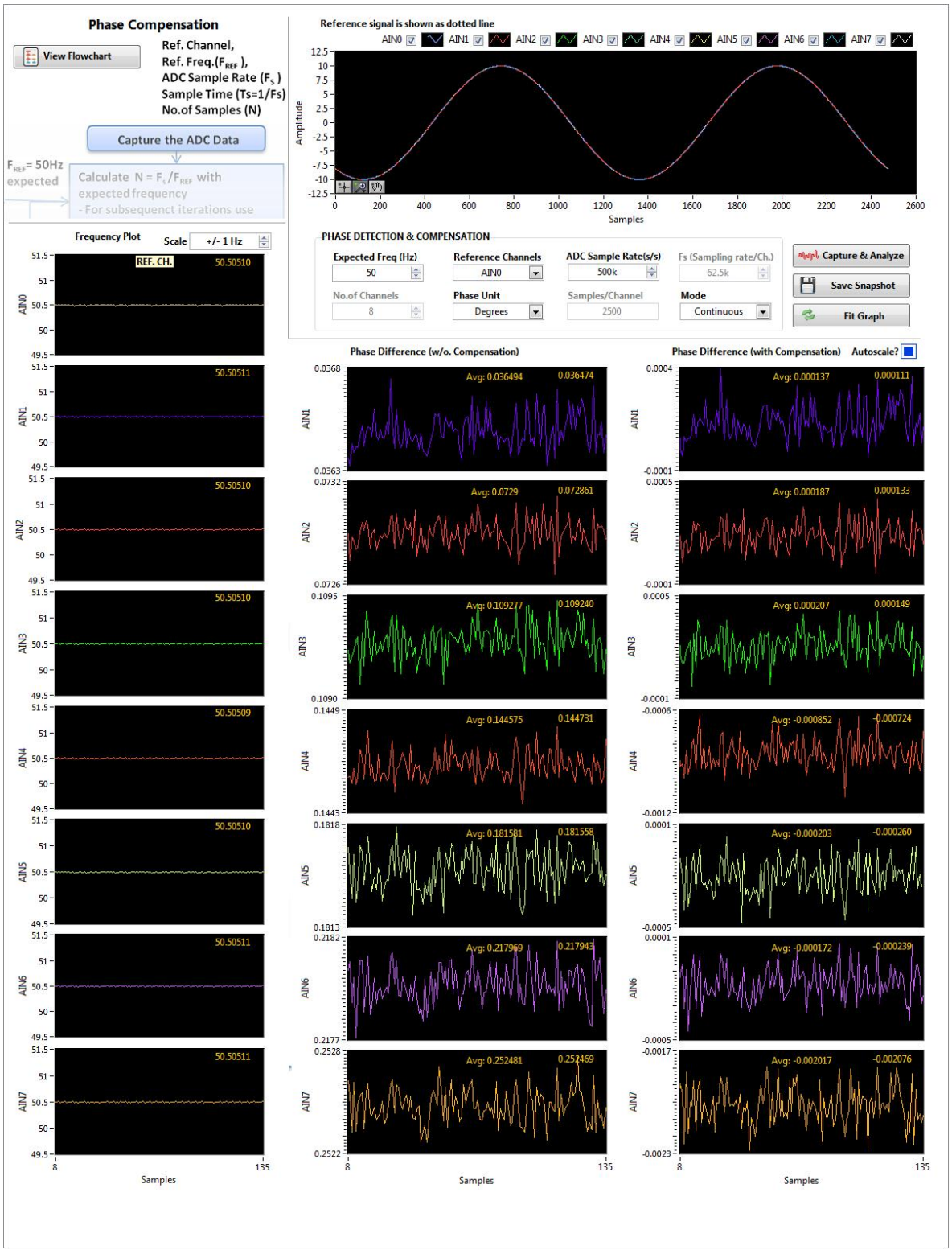


Figure 16. Screenshot of Verification Test 2

This test shows a similar result as Case 1 in Section 5.1. The test also demonstrates that the tested result matches the theoretically calculated values and the phase difference is still constant between two consecutive channels, despite a change in the signal frequency. However, the calculation error of the software increases because of the frequency difference between the actual and standard 50-Hz signal.

Table 4 shows the test result summary for Figure 16.

Table 4. Measured Phase Difference Summary Before and After Compensation—Case 2(1)

50.5-Hz SINUSOIDAL INPUT	UNCOMPENSATED PHASE DIFFERENCE Δθ	COMPENSATED PHASE DIFFERENCE Δθ
Phase difference (Ch1 – Ch0)	0.036494º	0.000137º
Phase difference (Ch2 – Ch0)	0.0729º	0.000187º
Phase difference (Ch3 – Ch0)	0.109277º	0.000207º
Phase difference (Ch4 – Ch0)	0.144575º	–0.000852º
Phase difference (Ch5 – Ch0)	0.181581º	–0.000203º
Phase difference (Ch6 – Ch0)	0.217969º	–0.000172º
Phase difference (Ch7 – Ch0)	0.252481º	–0.002017º

(1) Channel 0 is the reference channel for the test.

5.3 Verification Analysis—Case 3: All Signal Frequencies Same With Different Initial Phase Angles

In the third case, the input signal frequency is the same for all channels, but the initial phase angles are different. Theoretically, the final test result for phase difference between any two channels should be the sum of their initial phase angle difference plus the phase angle variation as a result of frequency variation. Another component of phase difference is added because of the multiplexed ADC input, but this fixed phase difference can be compensated by the software.

Test equipment is not readily available that can output eight distributed signals with the same frequency and independent configuration of the initial phase angle for every channel. However, the simulation test function of phase compensation integrated in the ADS8688EVM-PDK GUI software can be used to verify this case by navigating to *Smart App* → *Phase Compensation* in the *File* menu. The test function uses the data created by the phase compensation software for every channel and is easy to configure for setting up different values of the initial phase angle for each channel. Figure 17 shows the initial phase angle setup.

For this experiment, the sampling rate of the ADS8688 device has been set to 500 kSPS and the frequency of the sinusoidal waveform has been set to 50 Hz for all eight channels by using the *Capture & Analyze* button on the front page of the GUI. The initial phase angle for all channels are set in increments of 1º, starting from CH0 at 0º, CH1 at 1°, and correspondingly, CH7 to 7° (refer to Figure 17). After executing the simulation command by clicking the *OK* button, the phase difference between CH0 and CH1 is observed as 1.036° on the results waveform. This resulting phase difference includes the initial phase difference of 1º plus the additional phase difference of 0.036º because of the multiplexed input structure of the ADC (outlined in Case 1 in Section 5.1). Similarly, the phase difference between CH0 and CH2 is approximately equal to 2.072°, and so on. As Figure 18 shows, this additional phase difference as a result of the multiplexed ADC can be compensated by using the software.

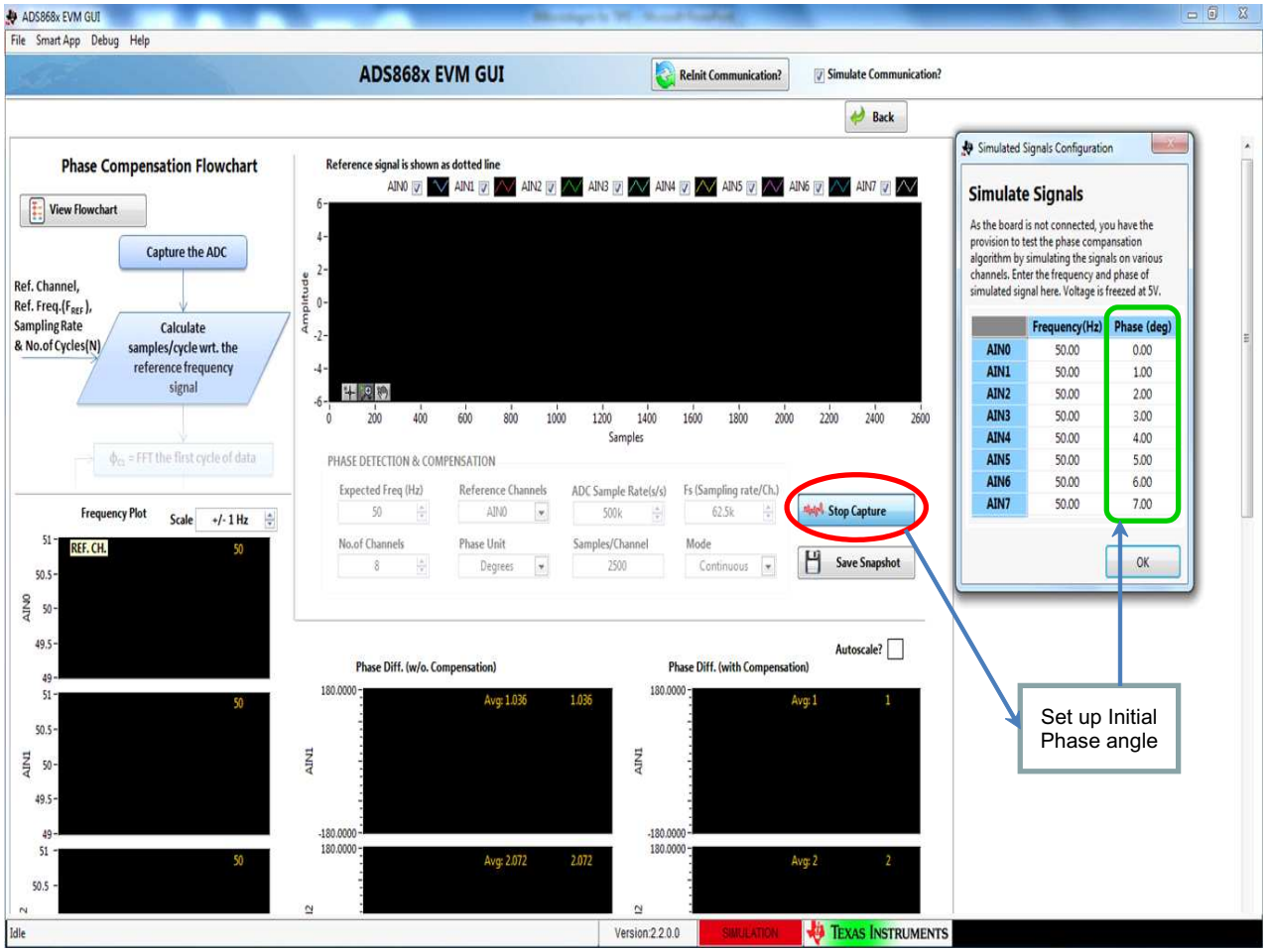


Figure 17. Phase Compensation With Initial Phase Angle Setup

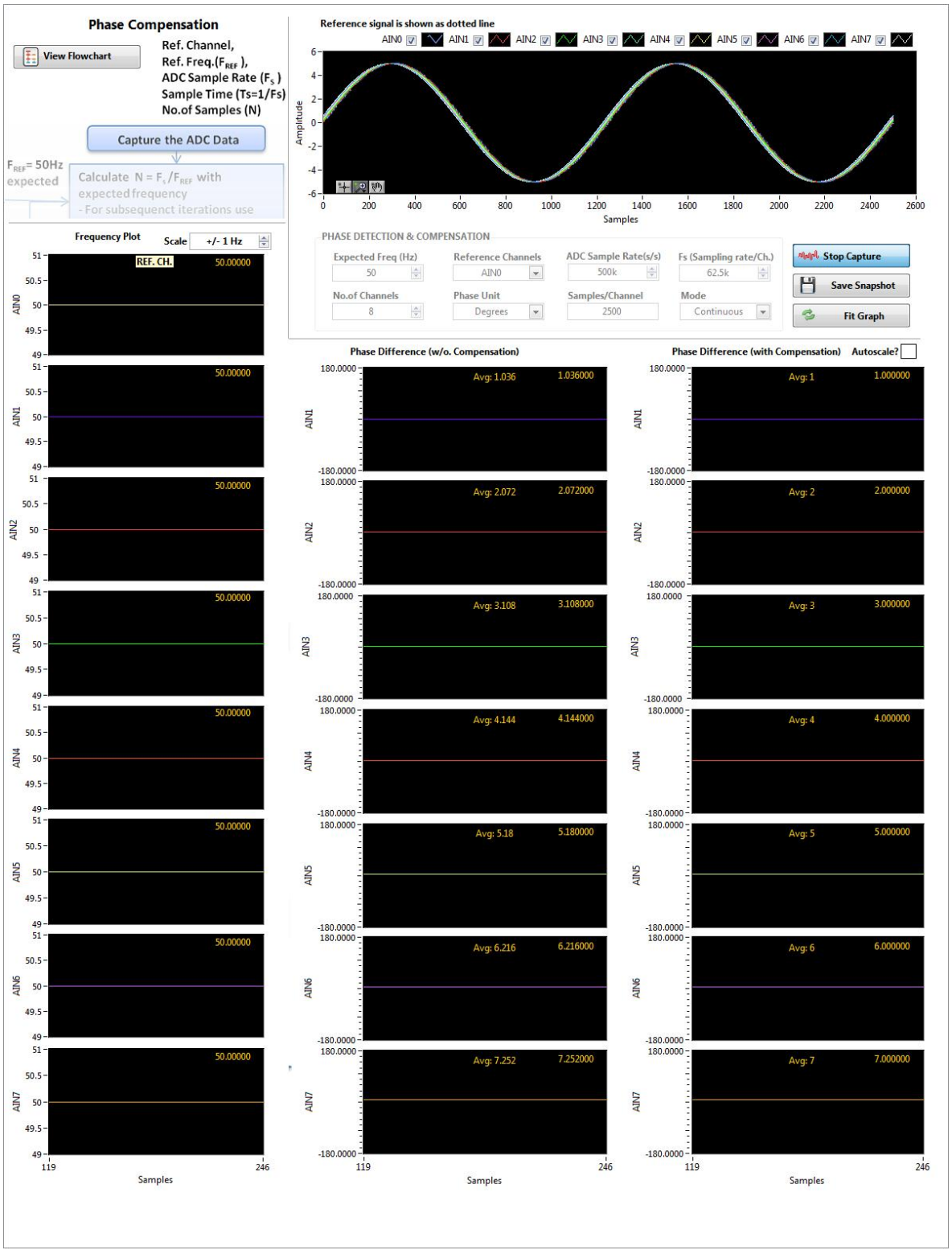


Figure 18. Screenshot of Verification Test 3 (Simulation)

5.4 Result Analysis

The tests in the preceding subsections show that the phase difference between different channels as a result of the multiplexed ADC ($\Delta\Phi_{mux}$) is constant and matches the theoretical calculations in Equation 15 and Equation 16, which also verify the phase difference compensation theory. The value of $\Delta\Phi_{mux}$ only depends on the sampling rate of the ADC and input signal frequency; so, $\Delta\Phi_{mux}$ can be subtracted from the measured total phase difference ($\Delta\Phi_{NSS}$) directly in the software according to the theoretical value. In other words, the phase compensation and calibration described in this TI Precision Design can easily be used for the input signals appearing at every channel. This feature makes the design very useful and ideal for use in power protection and measurement applications.

6 Modifications

The solution and theory introduced in this design are not only suitable for the ADS8688, but also suitable for other multiplexed input, SAR-based ADCs. One example is the ADS8332 device from TI, which is a low-power, 16-bit, 500-kSPS ADC with a unipolar, 8-to-1 multiplexer input. The hardware solution of the ADS8332 along with a precision amplifier from TI is another viable selection for performing phase compensation in power automation applications.

7 Design Files

7.1 Schematics

To download schematics, see the design files at TIPD167.

7.2 Bill of Materials

To download the bill of materials (BOM), see the design files at TIPD167.

7.3 Layer Plots

To download the layer plots, see the design files at TIPD167.

7.4 Layout Guidelines

7.4.1 PCB Layout

The primary emphasis for this TI Precision Design is the phase compensation software; however, the performance of the ADC is equally important as it affects the accuracy of the phase calculation in the software. Addresing the details of the PCB design is important to ensure the best performance from the ADS8688 device. Some of the key considerations and PCB layout guidelines for this design are as follows:

- Separate the analog signals from the digital signals:
  TI recommends to separate the analog and digital circuitry on the PCB by placing them in different planes or parts of the PCB. This separation also prevents tracks from crossing each other. The tracks carrying digital signals may introduce high frequency noise in analog signals because of coupling.
- Use separate power supplies for the analog and digital circuits:
  Implementing separate analog and digital power supplies is desirable. If using a switching-type power supply for the digital circuit, a separate linear supply should be used for the analog circuit. Implementing a separate supply for the analog section is preferable if a lot of noise results on the DC power supply (such as in digital I/O switching).
- Use separate PCB layers for the supply and ground:
  Wherever possible, try to use multilayered PCBs and use separate layers on the PCB for the power and ground. A full ground plane also provides a good shielding effect and reduces the electromagnetic induction susceptibility of the circuit. This design uses a total of six layers: two layers for the ground, one layer for the analog power, one layer for the digital power, one layer for the analog signal, and one layer the for digital signal.
- Reference path routing:
  A reference decoupling capacitor should be placed as close to the REFCAP and REFGND pins as possible. The REFGND pin should also be connected directly to the pad of this decoupling capacitor.
- Component location and routing:
  The input components should be located as close to the ADC input as possible to minimize trace and loop area and improve electromagnetic interference (EMI) and radio-frequency interferance (RFI) rejection. The trace for the power supply should be wide to avoid any voltage drop.

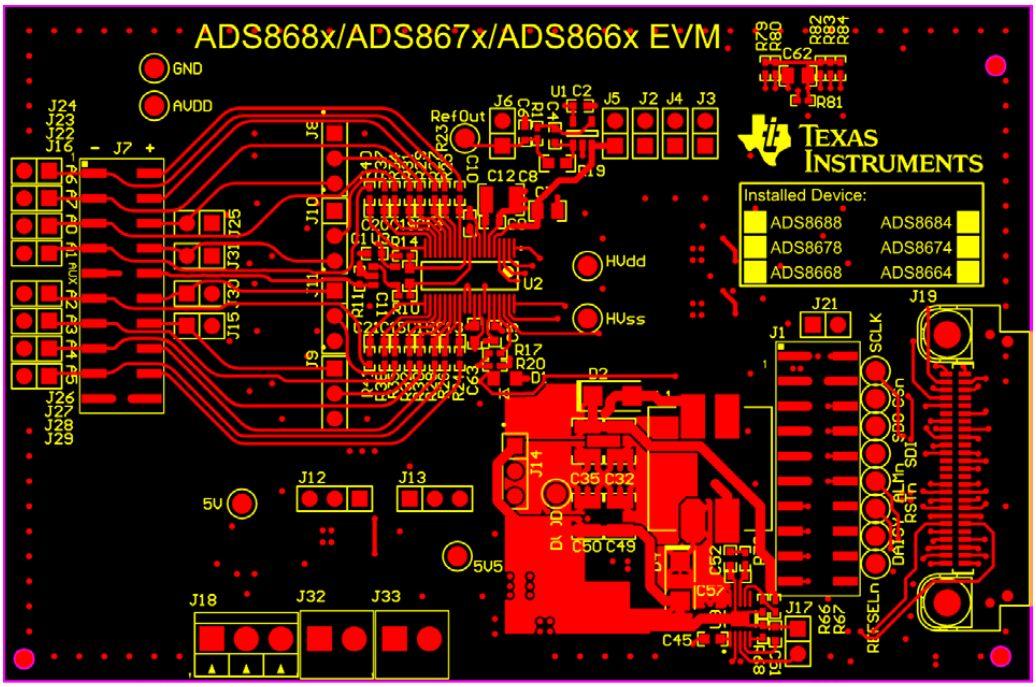


Figure 19. ADS8688EVM Top-Side Layout

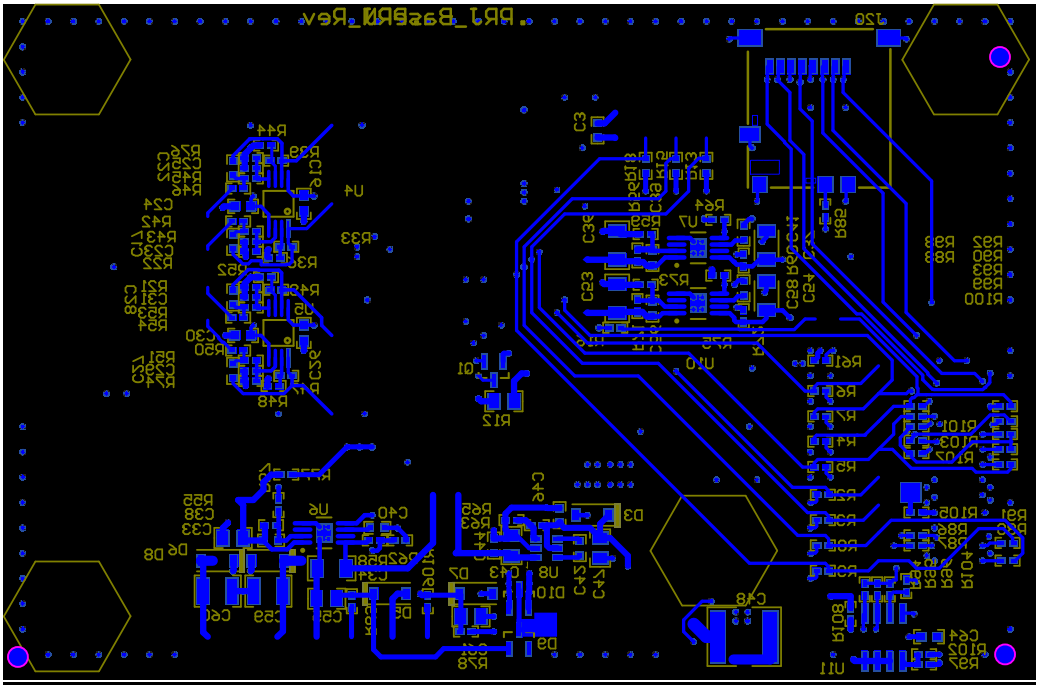


Figure 20. ADS8688EVM Bottom-Side Layout

7.5 Gerber Files

To download the Gerber files, see the design files at TIPD167.

7.6 Assembly Drawings

To download the assembly drawings, see the design files at TIPD167.

7.7 Software Files

To download the software files, see the design files at TIPD167.

8 Acknowledgments

Special thanks to Matthew Hann and Vinay Agarwal from Texas Instruments for their help and technical contributions to this design.

9 References

1. CNKI.com, Li Dedong, Wang Yi; *A new method for measuring phasor based on DFT and software-tracking*; Relay, July 16, 2006: (http://en.cnki.com.cn/Article_en/CJFDTotal-JDQW200614010.htm)
2. ResearchGate.net, Jiang Yaqun, He Yigang; *High-accuracy phase difference estimation between same frequency components in two periodic signals*; Transactions of China Electrotechnical Society, November 2006: (https://www.researchgate.net/publication/290771446_High-accuracy_phase_difference_estimation_between_same_frequency_components_in_two_periodic_signals)
3. CNKI.com, Mo Shaoqing, Li Xiaocong; *AN IMPROVED HIGH-ACCURACY ALGORITHM FOR FREQUENCY MEASUREMENT BASED ON FOURIER TRANSFORM*; Automation of Electric Power System, June 25, 2003: http://en.cnki.com.cn/Article_en/CJFDTOTAL-DLXT200312011.htm
4. Wikipedia.org, *Phase (waves)*, Wikipedia definition and electronics-tutorials (https://en.wikipedia.org/wiki/Phase_%28waves%29#Phase_difference)
5. Sheng Chunbo, Study on Phase Measurement in Power System, Yanshan University (http://wenku.baidu.com/view/9ded8bd7195f312b3169a5f6.html)
6. IEEE Xplore® Digital Library, A G Phadke, J S Thorp, M G Adamiak; *New Measurement Technique for Tracking Voltage Phasors, Local System Frequency, and Rate of Change of Frequency*; IEEE Transactions on Power Apparatus and Systems, May 5, 1983: (http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4112036)

10 About the Author

DALE LI is an Application Engineer in the Precision Analog, SAR ADC team at Texas Instruments based in Tucson, Arizona. Prior to this role, he has worked on Field Application Engineer at TI since 2007. Prior to joining TI, Dale was a Field Application Engineer and R&D engineer for 9 years and the covered industries include power automation and industrial instrument applications.

SENTHILNATHAN ARUMUGAM is a Test Automation Architect with several years of experience in measurement and automation software development using LabVIEW and PXI Instruments. He has architected several applications for automotive and semiconductor industry. Senthilnathan has earned his Bachelor of Engineering degree from PSG College of Technology, Coimbatore in India.

Appendix A Alternate Methods for Phase Measurement

A.1 Non-Fourier Transform Methods for Phase Measurement

A.1.1 Principle of Phase Measurement Based on First-Order Linear Interpolation

The first-order linear interpolation method measures the phase by finding the zero-crossing values of the two signals. As Figure 21 shows, the two signals are denoted as x(t) and y(t) respectively, which have the same frequency. The variables k and j represent the number of samples, x(t) passes through zero crossing from negative to positive between the number (k – 1) and k samples, y(t) passes through zero-crossing from negative to positive between the number (j – 1) and j samples.

The time difference between samples (k – 1) and k, as well as the difference between samples (j – 1) and j, is assumed to be very small; so, the angle difference between these sample sets is not too large for linear approximation of the sinusoidal waveform signal. Based on linear approximation processing, the slope between samples (k – 1) and k for signal x(t) can be calculated as, $(x_k – x_{k-1}) / T_s$, where T_s is the sampling time period. So the time, which is the time instant of x(t) between the zero-crossing from negative to positive and the k^{th} sample point, can be calculated as $x_k \times T_s / (x_k – x_{k-1})$. Similarly, $y_j \times T_s / (y_j – y_{j-1})$ is the time, which is the time instant of y(t) between the zero-crossing from negative to positive and the j^{th} sample point; so, the time interval between the zero-crossing points of the two signals is calculated in the following Equation 32:

$$\Delta t = \left(j - k + \frac{x_k}{(x_k - x_{k-1})} - \frac{y_j}{(y_j - y_{j-1})} \right) \bullet T_s \tag{32}$$

where

- x_k, x_{k-1} are the No. k and (k − 1) sampling values,
- y_j, y_{j-1} are the No. j and (j − 1) sampling values,
- T_s is the sampling period.

So the phase difference between the signals (T is the signal period) can be obtained in the following Equation 33:

$$\Phi = \Phi_x - \Phi_y = \frac{\Delta t}{T} \times 360^\circ = \left(j - k + \frac{x_k}{(x_k - x_{k-1})} - \frac{y_j}{(y_j - y_{j-1})} \right) \times \frac{T_s}{T} \times 360^\circ \tag{33}$$

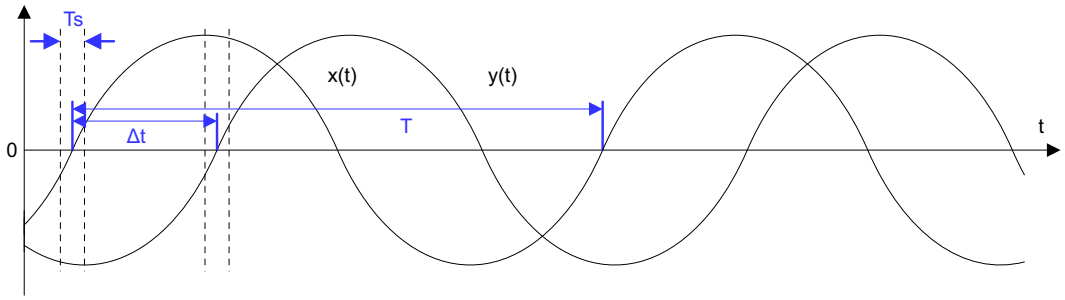


Figure 21. Calculation Chart of Phase Difference for First-order Linear Interpolation

Except for the first-order linear Interpolation method, the basic principle of using the quadratic interpolation method is often used for phase measurement and their principle is the same. Because the actual signal in power systems usually contains high frequency components, the first-order linear interpolation technique may not be accurate in judging the zero-crossover point; so, the quadratic interpolation method is used to reduce this calculation error by using a Lagrange interpolation polynomial.

A.1.2 Principle of Phase Measurement Based on Correlation Analysis Method

The correlation analysis method utilizes the principle of the cross-correlation between two sinusoidal signals with the same frequency at the zero time instant. The value of the cross-correlation function is proportional to the cosine of the phase difference between the two signals. Because the higher frequency harmonics or noise signal generally have little correlation with the valid signal, this method provides very good noise immunity. If there are two signals with the same frequency, denoted as *x*(t) and y(t), respectively, and both signals have a noise component, then the signal expressions are given as the following Equation 34:

$$\begin{aligned} x(t) &= A\sin(\omega_0 t + \Phi_0) + N_x(t) \\ y(t) &= B\sin(\omega_0 t + \Phi_1) + N_y(t) \end{aligned} \tag{34}$$

where

- A and B are the amplitudes of x(t) and y(t) respectively,
- $N_x(t)$ and $N_y(t)$ are noise signals,
- and $(\omega_0 t + \Phi_0)$ and $(\omega_0 t + \Phi 1)$ represent the phase of x(t) and y(t) respectively.

The correlation factor between the two signals can be calculated as shown in Equation 35:

$$\begin{aligned} R_{xy}(\tau) &= \frac{1}{T}\int_0^T \left(x(t)y(t+\tau)\right)dt \\ &= \frac{1}{T}\int_0^T \left[\left\{A\sin(\omega_0 t + \Phi_0) + N_x(t)\right\}\left\{B\sin(\omega_0(t+\tau) + \Phi_1) + N_y(t+\tau)\right\}\right]dt \end{aligned} \tag{35}$$

Substituting τ = 0 in Equation 35:

$$R_{xy}(0) = \frac{1}{T}\int_0^T \left[\left\{A\sin(\omega_0 t + \Phi_0) + N_x(t)\right\}\left\{B\sin(\omega_0 t + \Phi_1) + N_y(t)\right\}\right]dt \tag{36}$$

Because no correlation exists between the noise and actual signal, as well as the noise of the two signals, the above correlation function in Equation 36 can be simplified as:

$$\begin{aligned} R_{xy}(0) &= \frac{AB}{2}\cos(\Phi_1 - \Phi_0) \\ \Rightarrow \quad \Phi_1 - \Phi_0 &= \arccos\left(\frac{2 \times R_{xy}(0)}{AB}\right) \end{aligned} \tag{37}$$

Using this correlation principle for two independent signals, the expressions for amplitudes A and B can be calculated as $A = \sqrt{2 \times R_x(0)}$, $B = \sqrt{2 \times R_y(0)}$

Because the actual signal is the sampled sequence of a discrete point, the corresponding equations for the calculation of the discrete sequence are given by Equation 38:

$$\begin{aligned} R_{xy}(0) &= \frac{1}{k}\sum_{n=0}^{k-1} x(n)y(n) \\ R_x(0) &= \frac{1}{k}\sum_{n=0}^{k-1} x^2(n) \\ R_y(0) &= \frac{1}{k}\sum_{n=0}^{k-1} y^2(n) \end{aligned} \tag{38}$$

where

- k is the number of sample points.

The calculations for cross-correlation function of signals x(t) and y(t) as outlined in Equation 35 to Equation 38 can be used to calculate the phase difference between these two signals.

The correlation analysis technique can reduce any errors as a result of DC offset and noise interference between the concerned signals. The primary error in phase calculation is a result of the frequency instability of the AC signals. The correlation analysis method is extremely suitable for very-low frequency signals in comparison with the other phase measurement methods.

A.2 Asynchronous Sampling Compensation Method for Phase Measurement

This is an asynchronous method for the purposes of measuring the phase angle. The sampling frequency is fixed and does not require any additional hardware. The sampling frequency also does not require adjusting the size of the data window N to track the signal frequency. This method has the following features when it is used to phase difference measurement for two periodic signals with same frequency.

1. Does not require to know the exact signal frequency
2. Not necessary to guarantee the integer multiple relationship between the input signal and the sampling frequency.
3. Can measure the phase difference between two signals directly
4. Can measure the phase difference of each harmonic between different signals
5. Can suppress the spectral interference caused by side lobe well by using cosine window

A.2.1 Window Function

The FFT algorithm greatly improves the computation of the DFT; however, using recursive DFT algorithm for real-time calculation is much faster. The general recursive DFT algorithm is actually a recursive algorithm based on the rectangular window. Because the side-lobe peak of the window spectrum for a rectangular window is large, this algorithm does not suppress the spectral interference caused by the side lobe. Choosing a window function that reduces interference and spectral leakage can improve the accuracy of measurement.

The cosine window is popular to use, one example of which is the Blackman or Blackman-Harris window. Equation 39 shows the expression of a cosine window in a time domain:

$$w(n) = \sum_{h=0}^{H} a_h \cos\left(\frac{2\pi nh}{N}\right)$$
$$n = 0,1,\cdots,N-1 \tag{39}$$

where

- a_h is the coefficient of window,
- H is the term of polynomial of the coefficient.

Rectangular window: H = 0, a0 = 1

Blackman window: H = 2, a0 = 0.42, a0 = −0.5, a2 = 0.08

Blackman-Harris window: H = 3, a0 = 0.35875, a0 = −0.48829, a2 =0.14128, a3 = −0.01168

Equation 40 shows the expression of a cosine window in a time domain:

$$W(e^{j\omega}) = \sum_{h=0}^{H} \frac{a_h}{2}\left[W_R(\omega - h) + W_R(\omega + h)\right] \tag{40}$$

$$W_R(\omega) = \frac{\sin\left(\frac{N \times \omega}{2}\right)}{\sin\left(\frac{\omega}{2}\right)} \times e^{-j \times \frac{N-1}{2} \times \omega}$$

where

So:

$$W(e^{j\omega}) = \sin\left(\frac{N\omega}{2}\right)e^{-j\frac{N\omega}{2}}\left[\sum_{h=0}^{H}\frac{a_h}{2}\frac{\sin(\omega)}{\sin\left(\frac{\omega}{2}-\frac{\pi h}{N}\right)\sin\left(\frac{\omega}{2}+\frac{\pi h}{N}\right)} + j\sum_{h=0}^{H}a_h\right] \tag{41}$$

If meet $\sum_{h=0}^{H}a_h = 0\ (H \geq 1)$, then Equation 41 can be simplified as shown in the following Equation 42:

$$W\left(e^{j\omega}\right) = W(\omega)e^{-jC\omega} \tag{42}$$

where

- C is constant,
- C = N / 2

The result is Equation 43:

$$W(\omega) = \sin\left(\frac{N\omega}{2}\right)\sum_{h=0}^{H}\frac{a_h}{2}\frac{\sin(\omega)}{\sin\left(\frac{\omega}{2}-\frac{\pi h}{N}\right)\sin\left(\frac{\omega}{2}+\frac{\pi h}{N}\right)} \tag{43}$$

A.2.2 Phase Difference Measurement With Windowed DFT Method

Assuming two periodic signals with same frequency:

$$x_1(t) = \sum_{i=0}^{p}A_i\cos\left(2\pi i f_1 t + \alpha_i\right) \tag{44}$$

$$x_2(t) = \sum_{i=0}^{p}B_i\cos\left(2\pi i f_1 t + \beta_i\right) \tag{45}$$

where

- f_1 is the fundamental frequency of $x_1(t)$ and $x_2(t)$,
- A_i, ai, B_i, β_i are amplitude and phase (angle) for No. i harmonic of $x_1(t)$ and $x_2(t)$,
- and p is the highest harmonic.

The phase difference for No. i harmonic of $x_1(t)$ and $x_2(t)$ is $\theta_i = a_i - \beta_i$.

A.2.2.1 Phase Angle and Difference Calculation Analysis:

Calculate the actual α_i first. Discretize $x_1(t)$ using the sampling frequency, f_s, for which the following Equation 46 can be given:

$$x_1(n) = \sum_{i=0}^{p}A_i\cos\left(i * \omega_1 + \alpha_i\right) \tag{46}$$

where

- $\omega_1 = 2\pi \times f_1 / f_s$.

By adding a cosine window with length N, w(n), a finite discrete windowed sequence:

$$x_{1w}(n) = x_1(n)w(n)$$
$$0 \leq n \leq N-1 \tag{47}$$

Equation 48 calculates the spectrum for $x_{1w}(n)$:

$$X_{1w}(e^{j\omega}) = X_1(e^{j\omega})W(e^{j\omega}) = \sum_{i=0}^{p}\frac{A_i}{2}W(\omega - i\omega)e^{j\left[-C(\omega - i\omega_1)+\alpha_i\right]} + \sum_{i=0}^{p}\frac{A_i}{2}W(\omega + i\omega)e^{j\left[-C(\omega + i\omega_1)-\alpha_i\right]} \tag{48}$$

For No. i = l harmonic, when the cut-off time to signal is long enough and consider the effects of windowing, at $\omega = l\omega_1$ and close value, $W(\omega - i\omega_1)$ (when $i \neq l$) and $W(\omega + i\omega_1)$ are approximately zero, so when $\omega = i\omega = l\omega_1$,

$$X_{1w}(e^{j\omega}) = \frac{A_i}{2}W(\omega - i\omega)e^{j\left[-C(\omega - i\omega_1)+\alpha_i\right]} \tag{49}$$

Implementing DFT based on $x_{1w}(n)$, get discrete spectrum $X_{1w}(k)$. Actually $X_{1w}(k)$ is the sampling result basing on equivalent interval in range [0, 2π] for continuous spectrum, $X_{1w}(e^{j\omega})$.

$$X_{1w}(k) = X_{1w}(e^{j\omega})\Big|_{\omega=\frac{2\pi k}{N}}$$
$$k = 0,1,2,\ldots N-1 \tag{50}$$

If this is not integer period truncation in time domain, in other words, NT_s(T_s = 1 / f_s) (f_s is the sampling frequency) is not the integer multiple to T_1(T_1 = 1 / f_1), then

$$\frac{i\omega_1}{\Delta\omega} = \frac{i2\pi f_1 T_s}{\frac{2\pi}{N}} = \frac{iNT_s}{T_1} \tag{51}$$

This is not an integer maybe. Assuming

$$\frac{i\omega_1}{\Delta\omega} = k + \lambda \tag{52}$$

where

- k is a positive integer and $-0.5 \le \lambda \le 0.5$.

That is, the digital angular frequency of No. i harmonic is the closest to No. k spectrum line. From Equation 49 through Equation 52, calculate the spectrum of No. k spectrum line based on DFT as below:

$$X_{1w}(k) = \frac{A_i}{2}W(-\lambda\Delta\omega)e^{j\left(\frac{N\lambda\Delta\omega}{2}+\alpha_i\right)} = \frac{A_i}{2}W\left(-\frac{2\pi\lambda}{N}\right)e^{j(\pi\lambda+\alpha_i)} \tag{53}$$

So, the actual phase angle calculated by DFT is shown in Equation 54:

$$\alpha_i' = \alpha_i + \lambda\pi \tag{54}$$

Similarly, by adding the same cosine window to $x_2(t)$ and implementing DFT:

$$X_{2w}(k) = \frac{B_i}{2}W\left(-\frac{2\pi\lambda}{N}\right)e^{j(\pi\lambda+\beta_i)} \tag{55}$$

The corresponding phase angle is:

$$\beta_i' = \beta_i + \lambda\pi \tag{56}$$

So the phase difference for No. i harmonic between two signals is:

$$\theta_i = \alpha_i' - \beta_i' = \alpha_i - \beta_i \tag{57}$$

When $0 \neq \lambda$ and the sampling is non-integer period sampling, there is an error on the calculated phase angles (α_i' and β_i') by Equation 54 and Equation 56 for each signal; however, the error is equivalent and cancels out, so theoretically no error exists for the phase difference when sampling based on Equation 56. In other words, this lack of error demonstrates that this algorithm does not require integer period sampling or calculating to track the signal frequency.

A.2.2.2 Implementation Process:

After adding the window w(n) for the sequence $x_1(n)$, the real part and imaginary part of No. i harmonic (corresponding to the No. k spectrum line) can be calculated by DFT as follows in Equation 58 and Equation 59.

$$X_{1Re} = \frac{2}{N}\sum_{n=0}^{N} x_1(n)w(n)\cos\frac{2kn\pi}{N} \tag{58}$$

$$X_{1Im} = \frac{2}{N}\sum_{n=0}^{N} x_1(n)w(n)\sin\frac{2kn\pi}{N} \tag{59}$$

So, the phase angle for the $x_1(n)$ is $tg\left(\alpha_i'\right) = \frac{X_{1Im}}{X_{1Re}}$.

Similarly, for the sequence $x_2(n)$ in Equation 60:

$$X_{2Re} = \frac{2}{N}\sum_{n=0}^{N} x_2(n)w(n)\cos\frac{2kn\pi}{N}$$

$$X_{2Im} = \frac{2}{N}\sum_{n=0}^{N} x_2(n)w(n)\sin\frac{2kn\pi}{N} \tag{60}$$

The phase angle for the $x_2(n)$ is $tg\left(\beta_i'\right) = \frac{X_{2Im}}{X_{2Re}}$

$$tg\left(\alpha_i' - \beta_i'\right) = \frac{tg\left(\alpha_i'\right) - tg\left(\beta_i'\right)}{1 + tg\left(\alpha_i'\right)tg\left(\beta_i'\right)} = \frac{X_{1Im}X_{2Re} - X_{1Re}X_{2Im}}{X_{1Re}X_{2Re} + X_{1Im}X_{2Im}} \tag{61}$$

$$\theta_i = \alpha_i' - \beta_i' = tg^{-1}\left(\frac{X_{1Im}X_{2Re} - X_{1Re}X_{2Im}}{X_{1Re}X_{2Re} + X_{1Im}X_{2Im}}\right) \tag{62}$$

θ_i is the phase difference for No. i harmonic between two signals.

A.3 Hardware Synchronization Method

Figure 22 shows a typical block diagram of the hardware synchronization. The left-side component is the signal frequency monitor circuit and the right-side component is the frequency tracking (box with dotted lines), both of which change the sampling frequency according to the monitoring status. This hardware tracking circuit can also be replaced entirely with a software processing method for frequency tracking. After the power signal conditioning and filtering into a sinusoidal waveform within the range has completed, it is then converted to a square waveform after the zero-crossing detection and then the phase-locked loop (PLL) behind the detection circuit to track and lock the signal frequency and generates the sampling pulses to trigger an interrupt signal (ADC sampling control) to the CPU or FPGA. Therefore, the CPU or FPGA can control the ADC directly and change the sampling rate of the ADC to achieve the synchronous sampling. The PLL is mainly composed of a phase comparator (phase detector), a voltage controlled oscillator (VCO), and a low-pass filter. Because of the real-time tracking characteristics of the PLL, when the measured signal frequency f_i changes, the circuit can automatically track as fast and lock, which also always satisfies the relationship $f_0 = N \times f_i$, that is, the sampling frequency is an integer times (N) to the signal frequency. The N is determined by the number of samples; if the sample is 128 points per cycle, then N can be set to 128.

This method can achieve the synchronous sampling to eliminate the calculation errors as a result of asynchronous sampling, but this requires the specialized synchronization hardware circuit, which is more complex and increases costs.

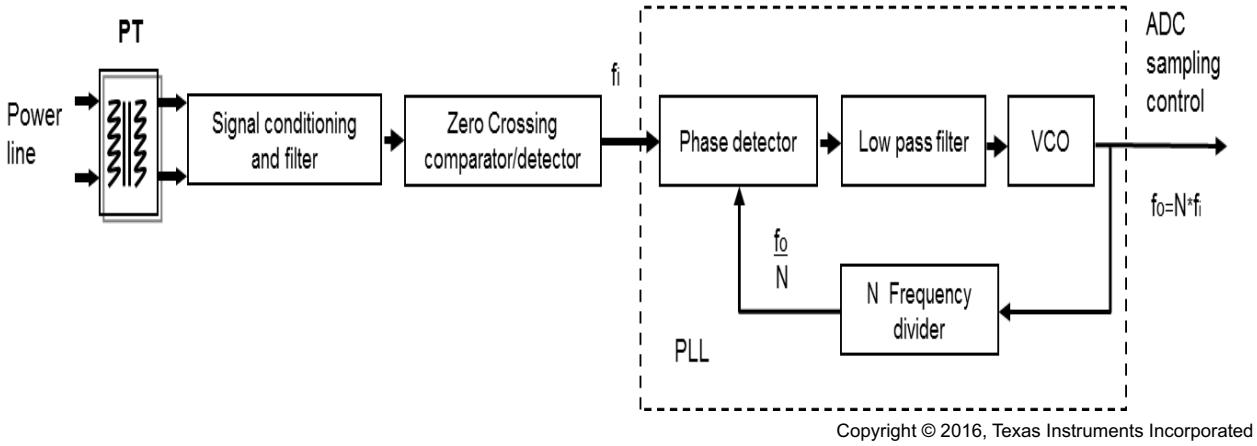


Figure 22. Hardware Synchronization for Synchronous Sampling

IMPORTANT NOTICE FOR TI REFERENCE DESIGNS

Texas Instruments Incorporated ('TI") reference designs are solely intended to assist designers ("Designer(s)") who are developing systems that incorporate TI products. TI has not conducted any testing other than that specifically described in the published documentation for a particular reference design.

TI's provision of reference designs and any other technical, applications or design advice, quality characterization, reliability data or other information or services does not expand or otherwise alter TI's applicable published warranties or warranty disclaimers for TI products, and no additional obligations or liabilities arise from TI providing such reference designs or other items.

TI reserves the right to make corrections, enhancements, improvements and other changes to its reference designs and other items.

Designer understands and agrees that Designer remains responsible for using its independent analysis, evaluation and judgment in designing Designer's systems and products, and has full and exclusive responsibility to assure the safety of its products and compliance of its products (and of all TI products used in or for such Designer's products) with all applicable regulations, laws and other applicable requirements. Designer represents that, with respect to its applications, it has all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. Designer agrees that prior to using or distributing any systems that include TI products, Designer will thoroughly test such systems and the functionality of such TI products as used in such systems. Designer may not use any TI products in life-critical medical equipment unless authorized officers of the parties have executed a special contract specifically governing such use. Life-critical medical equipment is medical equipment where failure of such equipment would cause serious bodily injury or death (e.g., life support, pacemakers, defibrillators, heart pumps, neurostimulators, and implantables). Such equipment includes, without limitation, all medical devices identified by the U.S. Food and Drug Administration as Class III devices and equivalent classifications outside the U.S.

Designers are authorized to use, copy and modify any individual TI reference design only in connection with the development of end products that include the TI product(s) identified in that reference design. HOWEVER, NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT OF TI OR ANY THIRD PARTY IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of the reference design or other items described above may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

TI REFERENCE DESIGNS AND OTHER ITEMS DESCRIBED ABOVE ARE PROVIDED "AS IS" AND WITH ALL FAULTS. TI DISCLAIMS ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING THE REFERENCE DESIGNS OR USE OF THE REFERENCE DESIGNS, INCLUDING BUT NOT LIMITED TO ACCURACY OR COMPLETENESS, TITLE, ANY EPIDEMIC FAILURE WARRANTY AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY DESIGNERS AGAINST ANY CLAIM, INCLUDING BUT NOT LIMITED TO ANY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON ANY COMBINATION OF PRODUCTS AS DESCRIBED IN A TI REFERENCE DESIGN OR OTHERWISE. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, DIRECT, SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF THE REFERENCE DESIGNS OR USE OF THE REFERENCE DESIGNS, AND REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

TI's standard terms of sale for semiconductor products (http://www.ti.com/sc/docs/stdterms.htm) apply to the sale of packaged integrated circuit products. Additional terms may apply to the use or sale of other types of TI products and services.

Designer will fully indemnify TI and its representatives against any damages, costs, losses, and/or liabilities arising out of Designer's non-compliance with the terms and provisions of this Notice.

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