Application Report

Compact Multichannel Current Sensing

ABSTRACT

This application report describes the implementation of a Class 1 energy measurement system that uses the TMCS1100 Hall-effect current sensors for compact, isolated current sensing. To obtain precise power and energy calculations, a ADS131M08 delta-sigma ADC, LM27762 charge pump, and a metrology caclulation microcontroller are used along with the TMCS1100.

This application report includes the necessary information with regard to the hardware for this implementation and the accuracy results obtained from this implementation.

Table of Contents

List of Figures

List of Tables

Trademarks

LaunchPad™ are trademarks of Texas Instruments.
All trademarks are the property of their respective owners.

1 Introduction

As the number of servers and other computer equipment in data centers drastically increases over time, being able to efficiently power multiple equipment becomes critical. To provide power to the large number of equipment in data centers, power distribution units (PDUs) are often used. Similar to a power strip, PDUs distribute the power at its input to its multiple outlets. Each outlet can be used to power a different server or other type of computer equipment. The current drawn from each outlet is measured and, along with the measured voltage, is used to calculate the power drawn from each outlet.

To maximize the number of outlets that can be implemented in a given form factor, it is imperative for the current sensing circuits of the outlet to be compact. Rogowski coils and current transformer current sensors are often relatively large, which in turn, leads to more area being occupied per outlet. An alternative option is to use shunt current sensors, which are relatively smaller than current transformers and Rogowski coils; however, shunts inherently do not have isolation, so they would require extra circuitry that increases the solution size if isolation is required. In addition, the shunt temperature would increase at higher currents. Due to the increase in temperature from operating at high currents and the heat generated by the servers themselves, the accuracy of the shunts could also drift as well since the shunt resistance drifts across temperature.

To address the limitations of the other potential PDU current sensors, a Hall-effect current sensor, such as the TMCS1100, can be used. The TMCS1100 is a galvanically isolated Hall-effect current sensor capable of DC or AC current measurement with temperature stability. The TMCS1100 Hall-effect sensor enables isolated, compact current sensing for PDU applications as well as other end equipment that may require compact current sensing, such as power quality meters.

For maximizing the accuracy of the power measurements, a high-precision ADC, such as the ADS131M08, should be used to measure the output of the TMCS1100 current sensor. The ADS131M08 device is an eight-channel, simultaneously-sampling, 24-bit, 2nd order delta-sigma (ΔΣ), analog-to-digital converter (ADC) that offers wide dynamic range. Using an ADC for the sensing and a separate microcontroller for the calculations provides flexibility when mapping channels. As an example, if it is desired to sense seven outlet currents and one input voltage, the ADS131M08 can support this by connecting one voltage sensing circuit to one ADC channel and seven current sensing circuits to the other seven channels. With a fixed function device that calculates the metrology parameters, typically only a maximum of four currents can be supported. As a result, at least two fixed function devices would be needed to sense one voltage and seven currents. In comparison, only one ADS131M08 device would be needed for this same scenario. Reducing the number of devices needed for PDUs that have a large number of sockets further reduces solution size and cost.

The processing in this design is done by the MSP432P4111, which acts as the metrology calculation microcontroller. This device has an Arm® 32-bit Cortex®-M4F CPU with Floating-Point Unit and Memory Protection Unit, a real-time clock, port mappable GPIOs, an AES encryption and decryption accelerator, and a CRC calculation module.

This application report describes how to use the TMCS1100 Hall-effect current sensor, ADS131M08 precision delta-sigma ADC, LM27762 charge pump, and a metrology calculation microcontroller to design a Class 1 energy measurement system. The results for an example implementation is also shown. The *System Specifications For Example Implementation* table shows the key system specifications of this example implementation.

Table 1-1. System Specifications For Example Implementation

FEATURES	DESCRIPTION
Selected current sensor	TMCS1100 Hall-effect current sensor
Selected ADC	ADS131M08
Selected microcontroller	MSP432P4111
Number of voltage and current channels	1 voltage and 3 current channels (ADC has 8 channels but only 4 channels of ADC used in this implementation)
Accuracy class	Class 1
Tested current range	0.1–20 A
Selected reference for ADC	Internal reference option for the ADS131M08 device
ADS131M08 Clock(CLKIN)	8,000,000 Hz derived from the 8.000 MHz crystal that is connected to the XTAL1 and XTAL2 pins of the ADS131M08 device
ADS131M08 Delta-sigma modulation clock frequency	4,000,000 Hz (= CLKIN / 2)
SPI Clock	8,192,000 Hz derived from 16.384-MHz crystal of the MSP432 (To support this frequency, the LaunchPad™ crystal was changed from 48 MHz to 16.384 MHz)
Oversampling ratio (OSR)	512
Digital filter output sample rate	7812.5 samples per second
Phase compensation implementation	Software
Phase compensation resolution	0.0090° at 50 Hz or 0.0108° at 60 Hz
Selected CPU clock frequency	48 MHz
System nominal frequency (f_{NOM})	50 or 60 Hz (selectable in software)
Metrology parameters measured	• 1-cycle root mean square (RMS) voltage for detecting sags, swells, and interruptions • 10 or 12 cycle RMS voltage, RMS current, fundamental RMS voltage, fundamental RMS current • Voltage underdeviation and voltage overdeviation • Phase to phase angle • Active, fundamental active, reactive, fundamental reactive, apparent , fundamental apparent power and energy • Power factor • Line frequency with zero crossing indication
Update rate for measured parameters	1-cycle for sag/swell RMS voltage readings; 10 cycles(when using 50-Hz nominal frequency) or 12 cycles(when using 60-Hz nominal frequency) for other parameters
Additional boards used for tested	• TIDA-00163 Isolated UART to RS-232 board for communication from LaunchPad to PC GUI • MSP-EXP432P4111 LaunchPad with its 48-MHz crystal replaced with a 16.384-MHz crystal
Board power supply	3.3-V output from MSP-EXP432P4111 LaunchPad

2 Implementation Block Diagram

The *Block Diagram* depicts the block diagram of the example TMCS1100 PDU implementation tested in this application note. This implementation uses the TMCS1100 Hall-effect current sensor, ADS131M08 delta-sigma ADC, the LM27762 charge pump, a metrology microcontroller, and an isolated UART to RS-232 board.

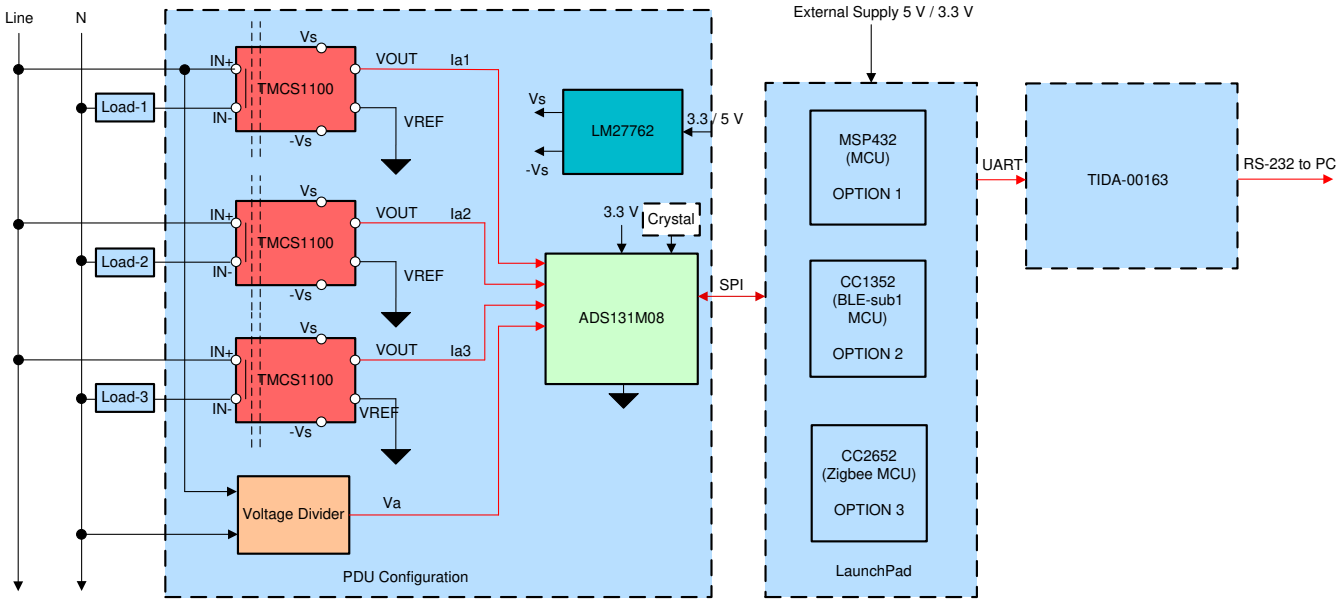

Figure 2-1. Block Diagram

In this implementation, there are four ADC channels that are used. One of these channels senses the inlet voltage. For sensing the voltage, a voltage divider divides the Mains voltage to a range that can be sensed by an ADC. The choice of voltage divider resistors for the voltage channel is selected to ensure the mains voltage is divided down to adhere to the normal input ranges of the ADS131M08 device. Since the ADS131M08 ADCs have a large dynamic range and a large dynamic range is not needed to measure voltage, the voltage front-end circuitry is purposely selected so that the maximum voltage seen at the inputs of the voltage channel ADCs are only a fraction of the full-scale voltage. By reducing the voltage fed to the ADS131M08 voltage ADC, voltage-to-current crosstalk, which actually affects metrology accuracy more than voltage ADC accuracy, is reduced at the cost of voltage accuracy, thereby resulting in more accurate energy measurements at lower currents.

The other three channels of this device are used to sense the current of the different outlets. Each current channel uses the TMCS1100 Hall-effect current sensor to translate the sensed current into a voltage sensable by the ADS131M08 ADC. To fit within the input voltage range of the ADS131M08, level shifting is necessary. This level shifting is accomplished by using the LM27762 charge pump to create 2.5-V and –2.5-V voltage rails that power the TMCS1100.

The ADS131M08 uses a crystal connected to its XTAL1 and XTAL2 pins to generate an internal clock, f_{CLKIN}. The ADS131M08 internally divides this clock by two and uses this divided down clock as the delta-sigma modulation clock, f_M. The sampling rate of the ADS131M08 is therefore defined as $f_s = f_M / OSR = f_{CLKIN} / (2 \times OSR)$. Whenever there are new samples available, the ADS131M08 asserts its $\overline{DRDY}$ pin to notify the microcontroller that new samples are available. The microcontroller would then use one of its SPI interfaces and its DMA to get the voltage and current samples from the ADS131M08 device. The microcontroller uses the new voltage and current samples for the calculation of the metrology parameters, such as the power and RMS readings.

The ADC connections to the microcontroller are brought out the LaunchPad connector of the design, which allows for different microcontrollers to be used as the metrology microcontroller by connecting the corresponding microcontroller LaunchPad to the LaunchPad connector of the design. For this specific implementation, the MSP432P4111 device was used as the metrology microcontroller by connecting the MSP-EXP432P4111 LaunchPad to the LaunchPad connector of the design.

For calibrating and testing the design, a PC GUI was used. The PC GUI communicates to the design through an isolated RS-232 connection created by the TIDA-00163 board. The TIDA-00163 board connects to a set of UART transmit (pin P2.5 on the MSP432) and receive (pin P2.3 on the MSP432) pins pin from the MSP432. The board isolates the signals from these pins and then translates the isolated signal to RS-232 signal levels. The resulting RS-232 signals are sent to the RS-232 connector of the TIDA-00163, which the PC is connected to.

3 Hardware Implemenation

3.1 Analog Inputs

Each of the ADS131M08 converters is differential and requires that the input voltages at its pins does not exceed ±1.2 V (assuming an internal PGA gain setting of 1 is used). To meet this input voltage specification, a voltage and current sensor circuit is needed to translate the Mains voltage and outlet current into a voltage that can be sensed by the ADS131M08. This subsection describes the analog front end used for voltage and current channels.

3.1.1 Voltage Measurement Analog Front End

The nominal voltage from the mains is from 100 V–240 V, so it needs to be scaled down to be sensed by an ADC. Since the ADS131M08 device can sense voltages down to 1.2 V, AC signals from mains can be divided down with a voltage divider and then fed to the ADS131M08 without the need for level shifters. In the board used for testing, there are four voltage sensing circuits; however, only the voltage divider circuit in the *Analog Front End for Voltage Inputs* figure is used for the PDU configuration referenced in this application note. The other three voltage sensing circuits are outside the scope of the PDU configuration referenced in this application note.

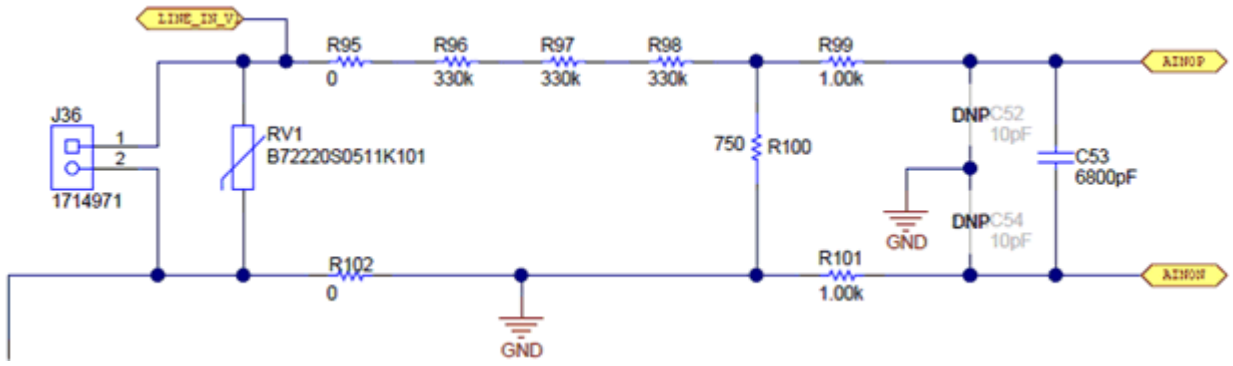

Figure 3-1. Analog Front End for Voltage Inputs

In the analog front end for voltage, J36 is where the input Mains voltage is applied. This circuit consists of a spike protection varistor (RV1), footprints for electromagnetic interference filter beads (resistor footprints R95 and R102) , a voltage divider network (R96, R97, R98, and R100), and an RC low-pass filter (R99, R101, and C53).

At lower currents, voltage-to-current crosstalk affects active energy accuracy much more than voltage accuracy if power offset calibration is not performed. To maximize the accuracy at these lower currents, the entire ADC range is not used for voltage channels. Since the ADCs of the ADS131M08 device are high-accuracy ADCs, using the reduced ADC range for the voltage channels in this design still provides more than enough accuracy for measuring voltage. Equation 1 shows how to calculate the range of differential voltages fed to the voltage ADC channel for a given Mains voltage and selected voltage divider resistor values:

$$V_{ADC\_Swing,Voltage} = \pm V_{RMS} \times \sqrt{2} \left(\frac{R100}{R96 + R97 + R98 + R100} \right) \quad (1)$$

Based on this formula and the previously-stated selected resistor values, for a mains voltage of 230 V, the input signal to the voltage ADC has a voltage swing of ±246 mV (174 mV_{RMS}). The ±246-mV voltage range is well within the ±1.2-V input voltage that can be sensed by the ADS131M08 device for the selected PGA gain value of 1 that is used for the voltage channels.

3.1.2 Current Measurement Analog Front End

The analog front end for current inputs is different from the analog front end for the voltage inputs. In this implementation, a TMCS1100 device is used for sensing each load current (3 devices in total). The A1 variant of the TMCS1100 is used, which has a sensitivity of 50 mV/A and allows sensing 20-A currents. The *TMCS1100 Current-Sensing Circuit* figure shows the TMCS1100 circuit used in this design.

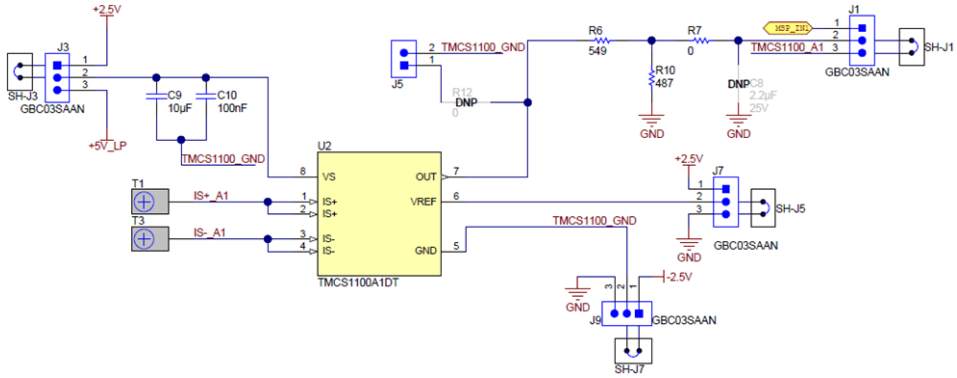

Figure 3-2. TMCS1100 Current-Sensing Circuit

In the circuit in the *TMCS1100 Current-Sensing Circuit* figure, the positive terminal of the input current is applied to T1 and the negative terminal of the input current is applied to terminal T3. Since terminal T1 is connected to pin IN+ (referred in the *TMCS1100 Current-Sensing Circuit* figure to as IS+) and terminal T3 is connected to pin IN- (referred in the *TMCS1100 Current-Sensing Circuit* figure as IS-), the current that flows through T1 and T3 also flows through the IN+ and IN- pins. The TMCS1100 works by sensing the current flowing through its set of IN+ and IN- pins.

When designing the front-end for the TMCS1100, the voltage between the power (pin VS) and GND pins of the device must meet the following constraint:

$$3 < \left(V_{TMCS1100\_VS} - V_{TMCS1100\_GND}\right) < 5.5 \quad (2)$$

In Equation 2, $V_{TMCS1100\_VS}$ is the voltage applied to the VS pin of the TMCS1100 device and $V_{TMCS1100\_GND}$ is the voltage applied to the GND pin of the TMCS1100.

In this design, 5 V is selected for the voltage between $V_{TMCS1100\_VS}$ and $V_{TMCS110\_GND}$, which meets the previously-listed criteria and also allows a wider current measurement range capability. Using this 5-V supply can lead to an output voltage that is larger than the input voltage range of the ADS131M08 ADCs. As a result, a voltage divider is needed to divide down the TMCS1100 output voltage to fall within the range of the ADC input range of the ADS131M08 device. R6 and R10 in the *TMCS1100 Current-Sensing Circuit* figure are selected to divide down the output voltage of the TMCS1100 to fit within the input ADC range of the ADS131M08 device. The R6 and R10 voltage divider results in the following input voltage at the ADC of the ADS131M08:

$$V_{ADS131M08\_IN} = \left(V_{TMCS1100\_OUT}\right)\left(\frac{R10}{R6 + R10}\right) = \left(S \times I_{In} + V_{TMCS1100\_Vref}\right)\left(\frac{R10}{R6 + R10}\right) \quad (3)$$

where,

- $V_{TMCS1100\_OUT}$ is the output voltage from the OUT pin of the TMCS1100 device
- S is the ideal sensitivity of the device (50 mV/A in this case)
- I_{IN} is the isolated input current provided to the inputs of the TMCS1100
- $V_{TMCS1100\_Vref}$ is the voltage at the VREF pin of the TMCS1100 device

The voltage range of the ADS131M08 is calculated by subtracting the minimum voltage the ADC can sense, $V_{ADS131M08\_IN,min}$, from the maximum voltage the ADC can sense, $V_{ADS131M08\_IN,max}$. Since the ADS131M08 has a

±1.2-V input voltage range when using a PGA gain of 1, $V_{ADS131M08\_IN,max}$ = 1.2 V, and $V_{ADS131M08\_IN,min}$ = –1.2 V. Therefore, the input voltage range of the ADS131M08 is 2.4 V.

The values of R6 and R10 should be selected so that the maximum output voltage range from the TMCS1100 fits within the input voltage range of the ADC of the ADS131M08.

The voltage across the OUT and GND pin of the TMCS1100 device will always be less than the voltage difference across its VS and GND pin. Due to this, a conservative set of resistor values for dividing down the voltage can be calculated using the following formula:

$$\left(\frac{R10}{R6 + R10}\right) < \left(\frac{V_{ADS131M08\_IN,max} - V_{ADS131M08\_IN,min}}{V_{TMCS1100\_VS} - V_{TMCS1100\_GND}}\right) = \left(\frac{2.4}{5}\right) \quad (4)$$

In this design, R6 was selected to be 549 Ω and R10 was selected to be 487 Ω; however, there are multiple options that can be selected for the total resistance value (R6 + R10) that would still meet Equation 4. Small values for this total resistance would increase the current drawn from the VOUT pin of the TMCS1100 device, which could limit the output voltage range of the TMCS1100. Conversely, large values for this series resistance could affect the anti-alias filter of the ADS131M08 device. In this design, the total resistance is around 1 kΩ , which does not reduce the output voltage range of the TMCS1100 and still does not significantly impact the anti-alias filter circuit for the ADS131M08.

Since the output voltage of the TMCS1100 is above 0 while half of the range of the ADS131M08 is below 0, a mechanism is needed to also level shift the voltage from the TMCS1100. This level shifting is done in this implementation by applying specific voltages to the VREF, VS, and GND pins of the TMCS1100 device.

The VREF pin on the TMCS1100 is used to set the voltage output of the TMCS1100 when applying 0 A. Setting VREF to the middle of the output swing range provides bidirectional measurement capability to sense both positive and negative currents. Therefore, Vref should ideally be set to the middle of the output voltage range of the TMCS1100, as expressed in the following equation:

$$V_{TMCS1100\_Vref} = \frac{V_{TMCS1100\_VS} + V_{TMCS1100\_GND}}{2} \quad (5)$$

To reduce DC offset in the current readings, the zero current condition should also provide an input voltage to the ADS131M08 that is at the mid-voltage point. As a result, the value of Vref should also ideally be close to meeting the following condition:

$$V_{TMCS1100\_Vref} = \left(\frac{V_{ADS131M08\_IN,max} + V_{ADS131M08_{IN},min}}{2}\right)\left(\frac{R6 + R10}{R10}\right) \quad (6)$$

Given that the voltage difference between the TMCS1100 Vs and GND pins is selected to be 5 V ($V_{TMCS1100\_VS}$ – $V_{TMCS1100\_GND}$ = 5), the two equations above can be solved to find the appropriate voltages that can be applied (with respect to the GND of the ADS131M08) on the TMCS1100 GND ($V_{TMCS1100\_GND}$), VS($V_{TMCS1100\_VS}$), and VREF ($V_{TMCS1100\_VREF}$) pins. For this design, the following voltages are used to meet this criteria:

- $V_{TMCS1100\_GND}$ = –2.5 V
- $V_{TMCS1100\_Vref}$ = –0 V
- $V_{TMCS1100\_VS}$ = 2.5 V

The 2.5 V and –2.5-V voltages are created by the LM27762 in the circuit shown in the *TMCS1100 GND and VS Voltage Generation Circuit* figure:

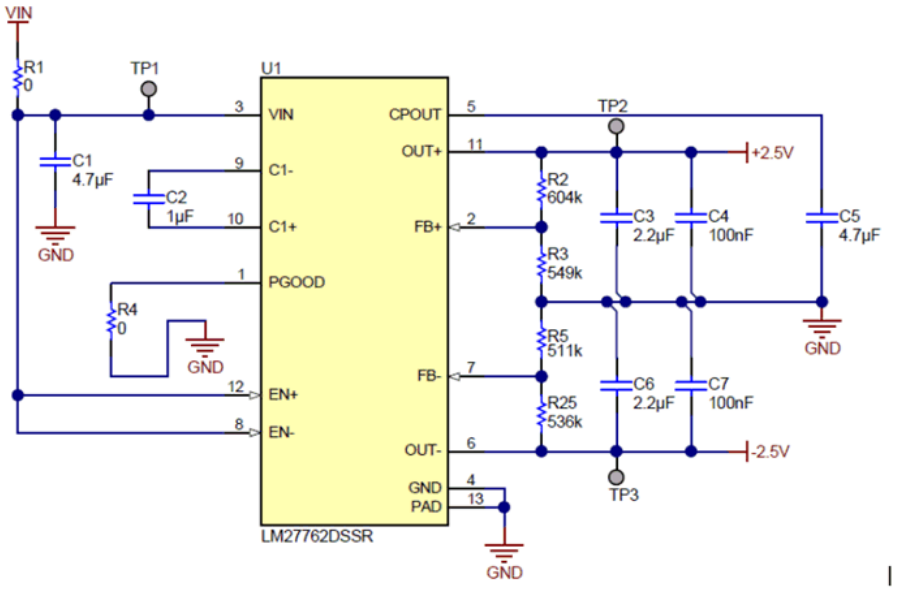

Figure 3-3. TMCS1100 GND and VS Voltage Generation Circuit

To apply these voltages to the TMCS1100 channel shown in the *TMCS1100 Current-Sensing Circuit* figure, jumpers should be placed between pins 1 and 2 of header J3, pins 2 and 3 of header J7, and pins 1 and 2 of header J9. The other jumper position options are for interfacing the TMCS1100 to alternative ADCs that only sense voltages above 0 V, such as the SAR ADC on the MSP432. In addition to these jumpers, a jumper should be placed between positions 2 and 3 of J1 to connect the TMCS1100 divided output to the circuit shown in the *ADS131M08 Interface Circuit for Current Channels* figure. The circuit in Figure 3-4 directly feeds into an ADC channel input of the ADS131M08.

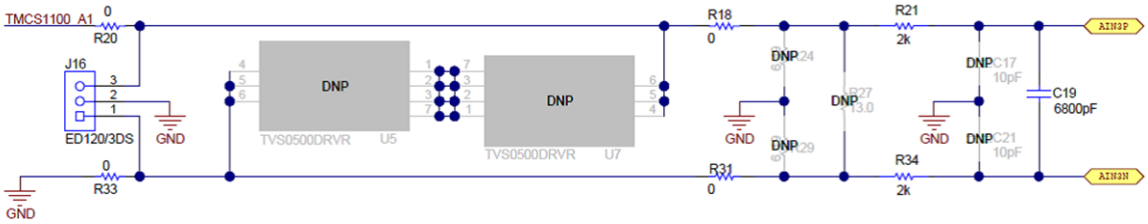

Figure 3-4. ADS131M08 Interface Circuit for Current Channels

The circuit in the *ADS131M08 Interface Circuit for Current Channels* figure consists of footprints for electromagnetic interference filter beads (R18 and R31), footprints (U5 and U7) that can be replaced with the TVS0500 for supplemental protection from surges, and an RC low-pass filter (R21, R34, and C19) that functions as an anti-alias filter. For the anti-alias filter, it should be noted that the R6 and R10 resistor values in the *TMCS1100 Current-Sensing Circuit* figure can affect the cutoff frequency of the anti-alias if the R6 and R10 total resistance is large; however, for the R6 and R10 resistance values used in this design, these resistor values do not significantly affect the cutoff frequency of the anti-alias filter.

To interface the differential inputs of the ADS131M08 with the single-ended output of the TMCS1100, the TMCS1100 divided output is connected to the positive terminal of the ADC channel of an ADS131M08 and the negative terminal of that ADS131M08 ADC channel is connected to the ground of the ADC. This results in the voltage at the positive terminal of the ADC swinging both above and below 0 while the voltage at the negative terminal of the ADC being fixed to the ADC GND voltage. Since the negative terminal of the ADC of the ADS131M08 is grounded, the input differential voltage to the ADC is also equal to the voltage at the positive

terminal of the ADC. The $V_{ADS131M08\_IN}$ Voltage Range figure shows the voltage range of $V_{ADS131M08\_IN}$ and how this range compares to the $V_{TMCS1100\_OUT}$ range, the $V_{TMCS1100\_VS}$ voltage, and the $V_{TMCS1100\_GND}$ voltage.

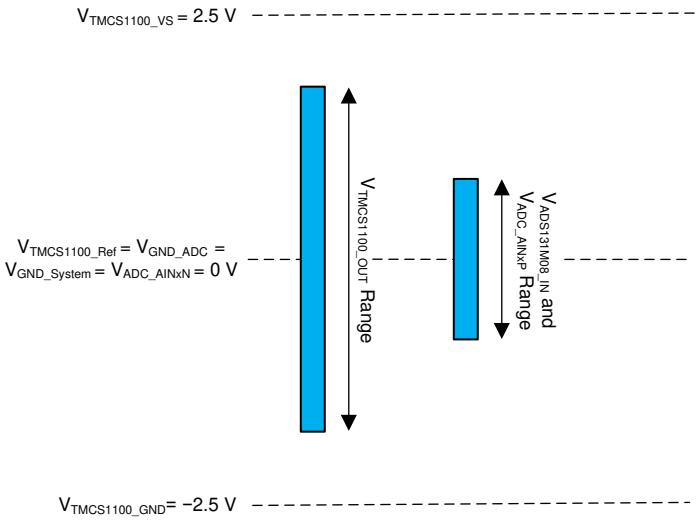

Figure 3-5. $V_{ADS131M08\_IN}$ Voltage Range

3.2 MSP432 LaunchPad™ Connections

This design has a LaunchPad connector that allows supporting connections to different microcontrollers. The *Implemented LaunchPad Connections* figure shows the implementation of the LaunchPad connector in this design:

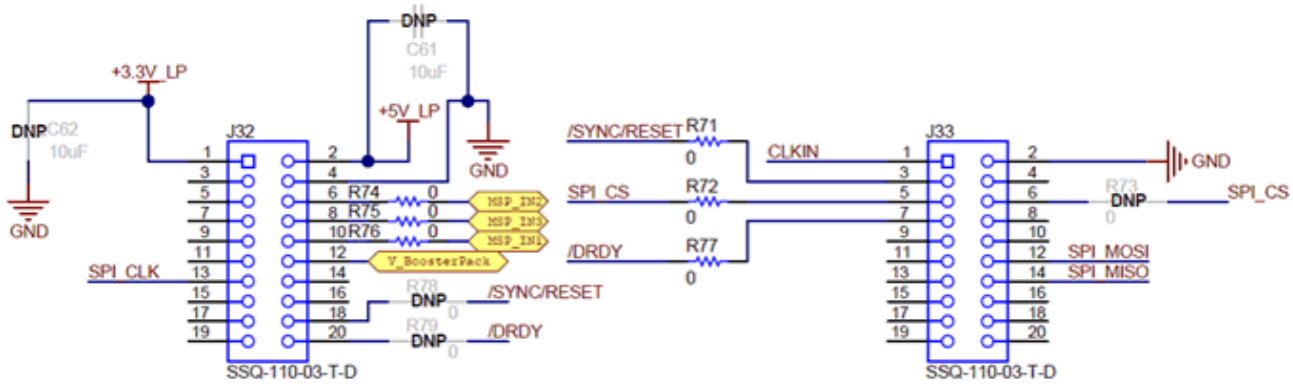

Figure 3-6. Implemented LaunchPad™ Connections

To support multiple microcontroller LaunchPads, there are two options for where to connect the SYNC/RESET (J32 pin 18 or J33 pin 3), SPI_CS (J33 pin 6 or J33 pin 5), and $\overline{DRDY}$ pins (J32 pin 20 or J33 pin 7). Each of these options have a resistor footprint that can be populated with a 0-Ω resistor for the options that are selected and depopulated for the options that are not selected.

For communication to the MSP432 LaunchPad specifically, the SYNC/RESET pin should be brought out to J33 pin 3, SPI_CS should be brought out to J33 pin 5, and $\overline{DRDY}$ should be brought out to J33 pin 7. These connections to the MSP432 LaunchPad is supported by populating resistors R71, R72, and R77 while depopulating resistors R78, R79, and R73, as is shown in the *Implemented LaunchPad Connections* figure.

The *LaunchPad Connection Mappings* table shows the connections that are used on the MSP432 LaunchPad for this design:

Table 3-1. LaunchPad™ Connection Mappings

LaunchPad™ PIN	FUNCTION
J33 pin 7 (P5.6)	ADS131M08 $\overline{\text{DRDY}}$
J33 pin 5 (P2.4)	ADS131M08 Chip Select
J33 pin 3 (P2.6)	ADS131M08 Reset/Sync
J33 pin 12 (P1.6)	ADS131M08 DIN (UCB0SIMO)
J32 pin 13 (P1.5)	ADS131M08 SPI Clock
J33 pin 14 (P1.7)	ADS131M08 DOUT(UCB0SOMI)
J33 pin 13 (P2.3)	EUSCIA0 UART RX for PC GUI communication
J33 pin 4 (P2.5)	EUSCIA0 UART RX for PC GUI Communication
J32 pin 15 (P4.6)	Active Energy Pulse Output
J32 pin 10 (P4.2)	Reactive Energy Pulse Output
J32 pin 14 (P4.5)	Voltage Zero Crossing Pin

3.3 PCB Layout Recommendations

- Refer to the layout guidelines in the TMCS1100, ADS131M08, and LM27662 data sheets.
- Note that the order of the AINxP and AINxN pins on the ADS131M08 switches when going from one ADC channel to another.
- Minimize the length of the traces used to connect the crystal to the ADS131M08. In addition, there must be clean ground underneath the crystal and placing any traces underneath the crystal must be avoided. Also, keep high-frequency signals away from the crystal.
- Use wide traces for power-supply connections.

4 How to Implement Software for Metrology Testing

The MSP432 software used for evaluating this design is test software. This section discusses the features of the test software, which should provide insights on how to implement custom software for metrology testing. The first subsection discusses the setup of the ADS131M08 device and various peripherals on the MSP432. Subsequently, the metrology software is described as two major processes: the foreground process and background process.

4.1 Setup

4.1.1 Clock

The MSP432 is configured to have its CPU clock (MCLK) set at 48 MHz and its subsystem master clock (SMCLK) set to 8.192 MHz. The clock source for MCLK is the internal DCO of the MSP432 MCU, which is configured for a frequency of 48 MHz.

The clock source for SMCLK is an external crystal. The MSP432 LaunchPad comes with a 48 MHz crystal (part number FA-238 48.0000MB-W0) by default, which is replaced with a 16.384 MHz crystal (part number: FA-238 16.3840MB-K). The new crystal also requires a 10-pF load capacitance instead of the 12-pF load capacitance of the previous crystal. To support this new crystal, the C11 and C12 22-pF crystal load capacitors were changed to 18-pF capacitors. The 16.384-MHz clock from the external crystal is internally divided by two to create the 8.192-MHz SMCLK frequency. Please note that the clock for the ADS131M08 is derived from its own 8.000-MHz crystal, so it is independent of the 8.192 MHz SMCLK of the MCU.

An external 32.768- kHz crystal is used as the clock source for the auxiliary clock (ACLK) of the device. This ACLK clock is set to a frequency of 32.768 kHz.

4.1.2 UART Setup for GUI Communication

The MSP432 MCU has a port mapping controller that allows a flexible mapping of digital functions to port pins. The set of digital functions that can be ported to other pins is dependent on the device. The port mapping controller is specifically used in this design to output the UART signals that communicate to the PCU GUI.

Using the port mapping controller, the following mappings are used:

- PMAP_UCA0RXD (EUSCIA0 UART RX) -> Port 2.3 (connected to the RX input of the TIDA-00163 board)
- PMAP_UCA0TXD(EUSCIA0 UART TX)->Port 2.5 (connected to the TX input of the TIDA-00163 board)

The MSP432 is configured to communicate to the PC GUI through the RS-232 connection on the TIDA-00163 board. The MSP432 communicates to the PC GUI using a UART module configured for 8N1 at 9600 baud.

4.1.3 Real-Time Clock (RTC)

The real-time clock module of the MSP432 is configured to give precise one-second interrupts and update the time and date, as necessary.

4.1.4 Direct Memory Access (DMA)

The direct memory access (DMA) module transfers packets between the MSP432 and ADS131M08 device with minimal bandwidth requirements from the MSP432 CPU. Two DMA channels are used for communicating to the ADS131M08. One channel (channel 0) is used to send data to the ADS131M08 and the other channel (channel 1) is used to receive data from the ADS131M08. Once a complete packet has been received from the ADS131M08, an interrupt is generated to complete any necessary post-transfer processing, such as CRC verification and packet assembly. The *ADS131M08 ADC Sample Request Packet* figure shows the packets that are sent and received using the DMA of the MSP432.

4.1.5 ADC Setup

The *ADC Initialization and Synchronization Process* figure shows the process used to initialize the ADS131M08 before conversions are started.

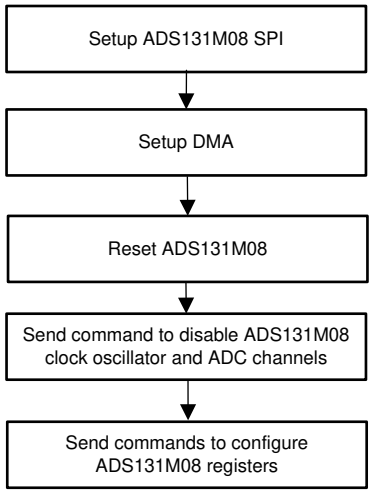

Figure 4-1. ADC Initialization and Synchronization Process

This process begins by the EUSCIB0 SPI module of the MSP432 MCU being configured for communication to the ADS131M08 device. The EUSCIB0 SPI module is specifically configured as a master device that uses 3-wire mode (the chip select signal is manually asserted high and low in the test software instead of using the chip select feature of the SPI module) and has an 8.192-MHz SPI clock that is derived from the 8.192-MHz SMCLK clock. After the SPI is setup, the DMA is then configured to automatically handle the transfer of packets between the ADS131M08 and MSP432.

Next, the ADS131M08 is reset to get the device in a known state. After resetting the device, a command is sent to the ADS131M08 to disable its internal oscillator and disable the conversions on its ADC channels. Disabling the clock oscillator prevents the ADS131M08 from generating new samples when configuring the ADS131M08 registers.

Finally, commands are sent to the ADS131M08 to initialize its registers for the following configuration:

- MODE register settings: 16-bit CCITT CRC used, 24-bit length for each word in the ADS131M08 packet, $\overline{\text{DRDY}}$ signal asserted on most lagging enabled channel, $\overline{\text{DRDY}}$ asserted high when conversion value is not available, $\overline{\text{DRDY}}$ asserted low when conversion values are ready
- GAIN1 and GAIN2 register settings: PGA gain of 1 used for all eight ADC channels (please note that only 4 of these 8 channels are used)
- CFG register settings: Current detection mode disabled
- CHx_CNG register settings (where x is the channel number): All eight ADC channel inputs connected to external ADC pins and channel phase delay set to 0 for each channel (note that software phase compensation is used instead of ADS131M08 hardware phase compensation)
- CLOCK register settings: 512 OSR, all channels enabled, crystal oscillator disabled, internal reference used for conversions, and high-resolution modulator power mode

At this point, the ADS131M08 is still not sampling because the crystal oscillator is still disabled. ADS131M08 sampling is not started until the rest of the setup routines of the MCU are executed. After executing these other setup routines, the MSP432 is configured to generate a port interrupt whenever a falling edge occurs on the $\overline{\text{DRDY}}$ pin, which would indicate that the ADS131M08 has new current samples that are available. Next, a command is sent to enable the crystal oscillator, which starts the voltage and current sampling.

The ADS131M08 modulator clock is derived from the crystal connected between its XTAL1 and XTAL2 pins. This clock is internally divided by two to generate the ADS131M08 modulator clock. The sampling frequency of the ADS131M08 is therefore defined as $f_s = f_M / OSR = f_{ADS13M08\_crystal} / (2 \times OSR)$, where f_s is the sampling rate, f_M is the modulator clock frequency, $f_{ADS13M08\_crystal}$ is the frequency of the crystal connected to the

ADS131M08, and OSR is the selected oversampling ratio. In this design, the crystal connected to the ADS131M08 has a frequency of 8.000 MHz and the oversampling ratio is selected to be 512. As a result, the ADS131M08 modulator clock is set to 4.000 MHz and the sample rate is set to 7812.5 samples per second.

In this design, the following ADS131M08 channel mappings are used in software:

- AIN0P and AIN0N ADS131M08 ADC channel pins → Voltage V1 (Line-to-Neutral Voltage)
- AIN5P and AIN5N ADS131M08 ADC channel pins → Current I1 (Channel A Current)
- AIN4P and AIN4N ADS131M08 ADC channel pins → Current I2 (Channel B Current)
- AIN3P and AIN3N ADS131M08 ADC channel pins → Current I3 (Channel C Current)

4.2 Foreground Process

The foreground process includes the initial setup of the MSP432 hardware and software and the ADS131M08 registers immediately after a device RESET. The *Foreground Process* figure shows the flowchart for this process. In the diagram, V-I mappings are referring to the combinations of the common voltage and the individual current channels. There are a total of three V-I mappings: the mapping between the voltage and current channel A, the mapping between the voltage and current channel B, and the mapping between the voltage and current channel C.

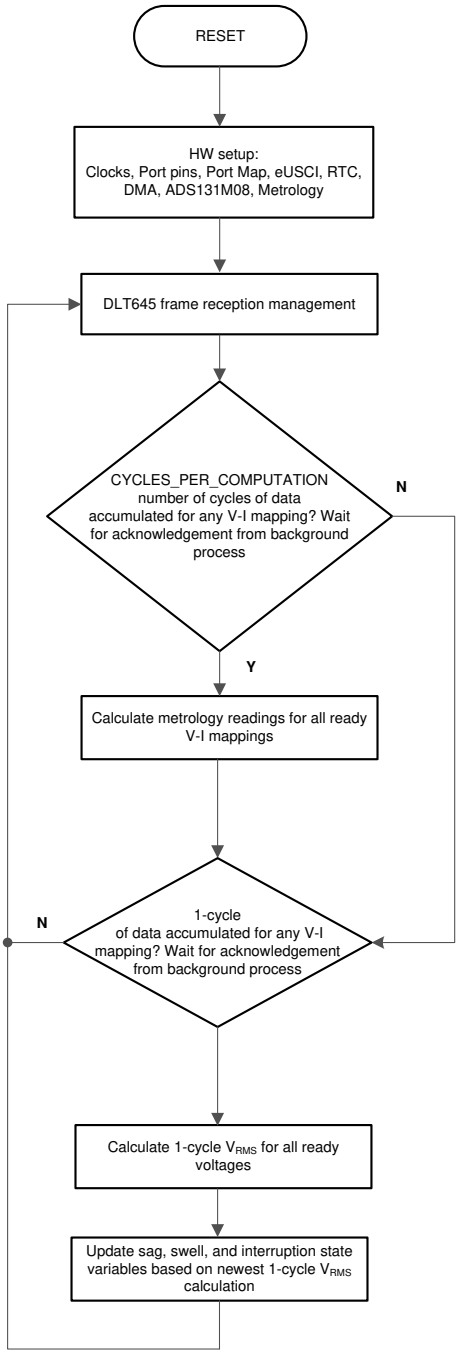

Figure 4-2. Foreground Process

The initialization routines involve the setup of the MSP432 clock system; general purpose input/output (GPIO) port pins and associated port map controller; MSP432 USCI_A0 for UART functionality; MSP432 RTC module for clock functionality; MSP432 DMA; ADS131M08 registers; and MSP432 metrology variables.

After the hardware is setup, any received frames from the GUI are processed. Next, the foreground process checks whether the background process has notified the foreground process to calculate new metrology parameters for any voltage-current mappings. This notification is accomplished through the assertion of the "PHASE_STATUS_NEW_LOG" status flag whenever a frame of data is available for processing. The data frame consists of the processed dot products that were accumulated for CYCLES_PER_COMPUTATION number of cycles of data. The value for CYCLES_PER_COMPUTATION is 10 cycles when the nominal frequency setting in the software is 50 Hz and 12 cycles when the nominal frequency setting in the software is set to 60 Hz. When the measured line frequency is equal to the nominal frequency of the design, this is equivalent to 200 milliseconds of accumulated data.

The processed dot products include the V_{RMS}, I_{RMS}, active power, reactive power, fundamental voltage, fundamental active power, and fundamental reactive power. These dot products are used by the foreground process to calculate the corresponding metrology readings in real-world units. All the processed dot products are accumulated in separate 64-bit registers to further process and obtain the RMS and mean values. Using the calculated values of active and reactive power of the foreground process, the apparent power is calculated. Similarly, using the calculated values of the foreground for the fundamental voltage, fundamental reactive power, and fundamental active power, the fundamental current, fundamental apparent power, voltage THD, and current THD are calculated. Additionally, voltage underdeviation and voltage overdeviation are calculated using the value of the calculated RMS voltage and the defined Nominal voltage of the design. The frequency (in Hz) and power factor are also calculated using parameters calculated by the background process using the formulas in the *Formulas* section.

The foreground process is also responsible for calculating the 1-cycle V_{RMS} readings that are used to update the sag, swell, and interruption state logging variables. The 1-cycle V_{RMS} readings are triggered by the background process after every negative to positive zero crossing. After the new V_{RMS} reading, the following state variables are updated accordingly:

- Swell variables
  - **Swell_events:** This variable logs the total number of swell events that have occurred since the design was first reset. The start of a swell event occurs whenever the 1-cycle RMS is above a user-defined sag. The end of a swell threshold event occurs when the 1-cycle RMS is below the user-defined swell threshold minus a user-defined hysteresis value.
  - **Max_swell_value:** This variable is the maximum 1-cycle RMS reading that was observed during the current ongoing swell event. If a swell event is not currently occurring, this variable represents the maximum 1-cycle RMS reading during the previously completed swell event.
  - **Swell_duration:** This variable logs the number of cycles during the current ongoing swell event. If a swell event is not currently occurring, this variable represents the duration of the previously completed swell event.
- Sag Variables
  - **Sag_events:** This variable logs the total number of sag events that have occurred since the design was first reset. The start of a sag event occurs whenever the 1-cycle RMS is below a user-defined sag threshold but is still greater than the user-defined interruption threshold. The end of a sag event occurs when the 1-cycle RMS is above the user-defined sag threshold plus a user-defined hysteresis value.
  - **Min_sag_value:** This variable is the minimum 1-cycle RMS reading that was observed during the current ongoing sag event. If a sag event is not currently occurring, this variable represents the minimum 1-cycle RMS reading during the previously completed sag event.
  - **Sag_duration:** This variable logs the number of cycles during the current ongoing sag event. If a sag event is not currently occurring, this variable represents the duration of the previously completed sag event.
- Interruption Variables
  - **Interruption_events:** This variable logs the total number of interruption events that have occurred since the design was first reset. The start of an interruption event occurs whenever the 1-cycle RMS is below a user-defined interruption threshold, which is lower than the sag voltage threshold.
  - **Interruption_duration:** This variable logs the number of cycles during the current ongoing sag event. If a sag event is not currently occurring, this variable represents the duration of the previously completed interruption event. Note that this variable does not increment when there is no voltage applied to the

design since there are no cycles to count; however, the absence of voltage is still able to be counted as an interruption event.

4.2.1 Formulas

4.2.1.1 Standard Metrology Parameters

This section briefly describes the formulas used for the voltage, current, power, and energy calculations. As previously described, voltage and current samples are obtained at a sampling rate of 7812.5 Hz. All of the samples that are taken in approximately 10 or 12 cycle frames are kept and used to obtain the RMS values for voltage and current for each phase.

The RMS, voltage overdeviation, and voltage underdeviation values are obtained with the following formulas:

$$V_{RMS,ph} = K_{v,ph} \times \sqrt{\frac{\sum_{n=1}^{\text{Sample Count}} V_{ph}(n) \times V_{ph}(n)}{\text{Sample Count}}} - V_{offset,ph} \tag{7}$$

$$V_{underdeviation,ph} = \begin{cases} 0, & \text{if } V_{RMS,ph} > V_{Nom} \\ \left(\frac{V_{Nom} - V_{RMS,ph}}{V_{Nom}} \right) \times 100, & \text{if } V_{RMS,ph} \leq V_{Nom} \end{cases} \tag{8}$$

$$V_{overdeviation,ph} = \begin{cases} 0, & \text{if } V_{RMS,ph} < V_{Nom} \\ \left(\frac{V_{RMS,ph} - V_{Nom}}{V_{Nom}} \right) \times 100, & \text{if } V_{RMS,ph} \geq V_{Nom} \end{cases} \tag{9}$$

$$I_{RMS,ph} = K_{i,ph} \times \sqrt{\frac{\sum_{n=1}^{\text{Sample Count}} I_{ph}(n) \times I_{ph}(n)}{\text{Sample Count}}} - I_{offset,ph} \tag{10}$$

where,

- ph = V-I mapping being calculated [that is, V-I_A (= 1) V-I_B (= 2) and V-I_C (= 3)]
- $V_{ph}(n)$ = Voltage sample at a sample instant n
- $V_{offset,ph}$ = Offset used to subtract effects of the additive white Gaussian noise from the voltage converter. This is in units of mV.
- V_{Nom}= Defined nominal voltage of design
- $I_{ph}(n)$ = Each current sample at a sample instant n
- $I_{offset,ph}$ = Offset used to subtract effects of the additive white Gaussian noise from the current converter. This is in units of µA.
- Sample count = Number of samples within the present frame
- $K_{v,ph}$ = Scaling factor for voltage
- $K_{i,ph}$ = Scaling factor for current

Power and energy are calculated for active and reactive energy samples of one frame. These samples are phase-corrected and passed on to the foreground process, which uses the number of samples (sample count) to calculate phase active and reactive powers through the following formulas:

$$P_{ACT,ph} = \left(K_{ACT,ph} \times \frac{\sum_{n=1}^{\text{Sample Count}} V_{ph}(n) \times i_{ph}(n)}{\text{Sample Count}} \right) - P_{ACT\_offset,ph} \tag{11}$$

$$P_{REACT,ph} = \left(K_{REACT,ph} \times \frac{\sum_{n=1}^{Sample\ Count} V_{90,ph}(n) \times I_{ph}(n)}{Sample\,Count} \right) - P_{REACT\_offset,ph} \quad (12)$$

$$P_{APP,ph} = \sqrt{P_{ACT,ph}^{2} + P_{REACT,ph}^{2}} \quad (13)$$

Where,

- $V_{90}(n)$ = Voltage sample at a sample instant 'n' shifted by 90°
- $K_{ACT,ph}$ = Scaling factor for active power
- $K_{REACT,ph}$ = Scaling factor for reactive power
- $P_{ACT\_offset,ph}$ = Offset used to subtract effects of crosstalk on the active power measurements from other currents
- $P_{REACT\_offset,ph}$ = Offset used to subtract effects of crosstalk on the reactive power measurements from other currents

Note that for reactive energy, the 90° phase shift approach is used for two reasons:

1. This approach allows accurate measurement of the reactive power for very small currents

2. This approach conforms to the measurement method specified by IEC and ANSI standards

The calculated mains frequency is used to calculate the 90 degrees-shifted voltage sample. Because the frequency of the mains varies, the mains frequency is first measured accurately to phase shift the voltage samples accordingly.

To get an exact 90° phase shift, interpolation is used between two samples. For these two samples, a voltage sample slightly more than 90 degrees before the most recent voltage sample and a voltage sample slightly less than 90 degrees before the most recent voltage sample are used. The phase shift implementation of the application consists of an integer part and a fractional part. The integer part is realized by providing an N samples delay. The fractional part is realized by a one-tap FIR filter. In the test software, a lookup table provides the filter coefficients that are used to create the fractional delays.

Using the calculated powers, energies are calculated with the following formulas :

$$E_{ACT,ph} = P_{ACT,ph} \times Sample\,Count \quad (14)$$

$$E_{REACT,ph} = P_{REACT,ph} \times Sample\,Count \quad (15)$$

$$E_{APP,ph} = P_{APP,ph} \times Sample\,Count \quad (16)$$

The calculated energies are then accumulated into buffers that store the total amount of energy consumed since system reset. Note that these energies are different from the working variables used to accumulate energy for outputting energy pulses. There are three sets of buffers that are available: one for each V-I mapping. Within each set of buffers, the following energies are accumulated:

1. Active import energy (active energy when active power ≥ 0)
2. Active export energy (active energy when active power < 0)
3. Fundamental active import energy (fundamental active energy when fundamental active power ≥ 0)
4. Fundamental active export energy (fundamental active energy when fundamental active power < 0)
5. React. Quad I energy (reactive energy when reactive power ≥ 0 and active power ≥ 0; inductive load)
6. React. Quad II energy (reactive energy when reactive power ≥ 0 and active power < 0; capacitive generator)
7. React. Quad III energy (reactive energy when reactive power < 0 and active power < 0; inductive generator)
8. React. Quad IV energy (reactive energy when reactive power < 0 and active power ≥ 0; capacitive load)
9. App. import energy (apparent energy when active power ≥ 0)
10. App. export energy (apparent energy when active power < 0)

The background process also calculates the frequency in terms of samples-per-mains cycle. The foreground process then converts this samples-per-mains cycle to Hertz with Equation 17:

$$\text{Frequency}(\text{Hz}) = \frac{\text{Sample Rate}(\text{in units of samples per second})}{\text{Frequency}(\text{in units of samples per cycle})} \tag{17}$$

After the active power and apparent power have been calculated, the absolute value of the power factor is calculated. In the internal representation of power factor of the system, a positive power factor corresponds to a capacitive load; a negative power factor corresponds to an inductive load. The sign of the internal representation of power factor is determined by whether the current leads or lags voltage, which is determined in the background process. Therefore, the internal representation of power factor is calculated with the following equation:

$$\text{Internal Representation of Power Factor} = \begin{cases} \frac{P_{ACT}}{P_{App}}, & \text{if capacitive load} \\ -\frac{P_{ACT}}{P_{App}} & \text{if inductive load} \end{cases} \tag{18}$$

4.2.1.2 Power Quality Formulas

For calculating the fundamental RMS voltage, a pure sine wave is generated and tightly locked to the fundamental of the incoming voltage waveform. Using the generated waveform, the fundamental voltage, fundamental active power, and fundamental reactive power are calculated by the following equations:

$$V_{fund,ph} = K_{v\_fund,ph} \times \frac{\sum_{n=1}^{\text{Sample Count}} V_{pure,ph}(n) \times V_{ph}(n)}{\text{Sample Count}} - V_{fund\_offset,ph} \tag{19}$$

$$P_{ACT\_fund,ph} = \left(K_{ACT\_fund,ph} \times \frac{\sum_{n=1}^{\text{Sample Count}} V_{pure,ph}(n) \times i_{ph}(n)}{\text{Sample Count}} \right) - P_{ACT\_fund\_offset,ph} \tag{20}$$

$$P_{REACT\_fund,ph} = \left(K_{REACT\_fund,ph} \times \frac{\sum_{n=1}^{\text{Sample Count}} V_{90\_pure,ph}(n) \times I_{ph}(n)}{\text{Sample Count}} \right) - P_{REACT\_fund\_offset,ph} \tag{21}$$

Where,

- $V_{pure,ph}(n)$ = Voltage sample of the pure sine wave generated, taken at a sample instant *n*
- $V_{90\_pure,ph}(n)$ = Voltage sample of the waveform that results from shifting $V_{pure,ph}(n)$ by 90° , taken at a sample instant n
- $K_{v\_fund,ph}$ = Scaling factor for fundamental voltage
- $K_{ACT\_fund,ph}$ = Scaling factor for fundamental active power
- $K_{REACT\_fund,ph}$ = Scaling factor for fundamental active power
- $V_{fund\_offset\_,ph}$= Offset to subtract from fundamental voltage calculation. This is in units of mV.
- $P_{ACT\_fund\_offset,ph}$ = Offset to subtract from fundamental active power calculation. This is in units of mW.
- $P_{REACT\_fund\_offset,ph}$ = Offset to subtract from fundamental reactive power calculation. This is in units of mvar.

After calculating the fundamental voltage, fundamental active power, and fundamental reactive power, the fundamental current and fundamental apparent power are calculated by the following formula:

$$I_{fund,ph} = \left(K_{i_{fund},ph} \times \frac{\sqrt{P^2_{ACT_{fund},ph} + P^2_{REACT_{fund},ph}}}{V_{fund,ph}} \right) - I_{fund\_offset,ph} \quad (22)$$

$$P_{APP\_fund,ph} = \sqrt{P^2_{ACT_{fund},ph} + P^2_{REACT_{fund},ph}} \quad (23)$$

Where,

- $K_{i\_fund,ph}$ = Scaling factor for fundamental current
- $I_{fund\_offset\_,ph}$= Offset to subtract from fundamental current calculation. This is in units of µA.

Once the fundamental current and fundamental voltage are calculated, the voltage THD and current THD can also be calculated. This software supports three different methods of calculating THD that are referred to as THD_{IEC_F}, THD_{IEC_R} , and THD_{IEEE}. The formulas used to calculate voltage THD (V_THD) and current THD (I_THD) with the different methods as follows:

$$V\_THD_{IEC\_F,ph} = \frac{\sqrt{V^2_{RMS,ph} - V^2_{fund,ph}}}{V_{fund,ph}} \quad I\_THD_{IEC\_F,ph} = \frac{\sqrt{I^2_{RMS,ph} - I^2_{fund,ph}}}{I_{fund,ph}} \quad (24)$$

$$V\_THD_{IEC\_R,ph} = \frac{\sqrt{V^2_{RMS,ph} - V^2_{fund,ph}}}{V_{RMS,ph}} \quad I\_THD_{IEC\_R,ph} = \frac{\sqrt{I^2_{RMS,ph} - I^2_{fund,ph}}}{I_{RMs,ph}} \quad (25)$$

$$V\_THD_{IEEE,ph} = \frac{V^2_{RMS,ph} - V^2_{fund,ph}}{V^2_{fund,ph}} \quad I\_THD_{IEEE\_F,ph} = \frac{I^2_{RMS,ph} - I^2_{fund,ph}}{I^2_{fund,ph}} \quad (26)$$

To calculate THD correctly, it is necessary to select the proper method of THD calculation and to ensure that any reference meter used for measuring THD uses the same THD method as the method selected in software.

4.3 Background Process

The *Voltage and Current Sampling Events* figure shows the different events that occur when sampling voltage and current, where the items in olive green are done by the hardware settings and not the test software.

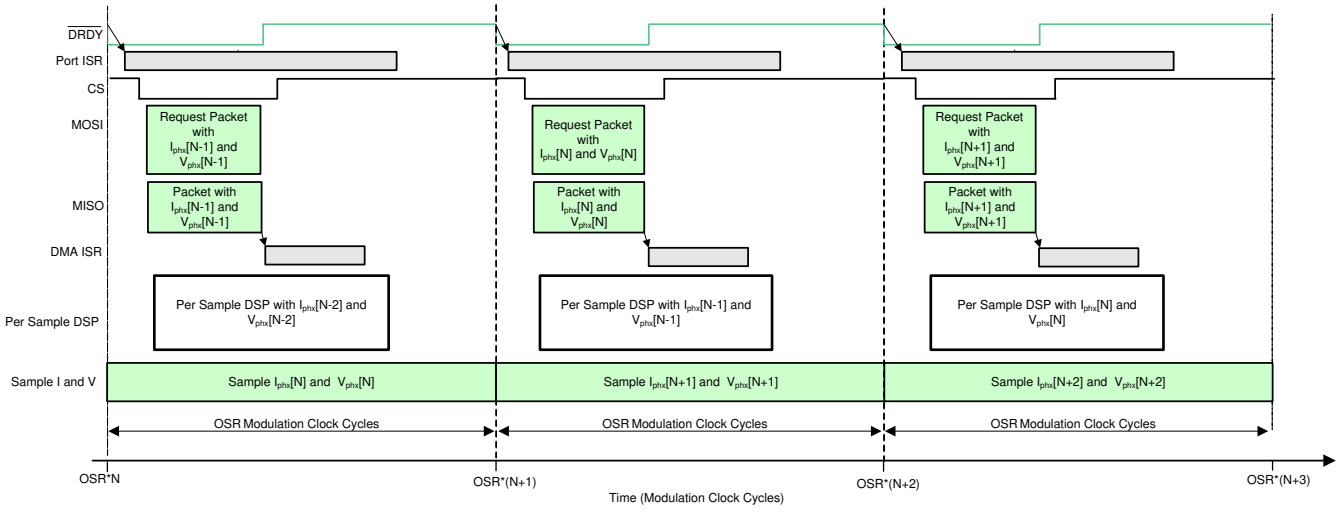

Figure 4-3. Voltage and Current Sampling Events

To go over the process mentioned in the *Voltage and Current Sampling Events* figure, new current samples for each phase are ready every OSR, or 512 for this design, modulation clock cycles. Suppose the most recently

ready phase current and voltage samples from the ADS131M08 device corresponds to the Nth – 1 current and voltage sample, or $I_{phx}[N – 1]$ and $V_{phx}[N – 1]$. Once new samples are ready, the $\overline{DRDY}$ pin is asserted low by the ADS131M08. The falling edge on the $\overline{DRDY}$ pin on the ADS131M08 causes a GPIO port interrupt on the MSP432, which triggers the Port ISR on the MSP432. The background process is run within the Port ISR. The *Background Process* figure shows the background process, which mainly deals with timing critical events in the test software.

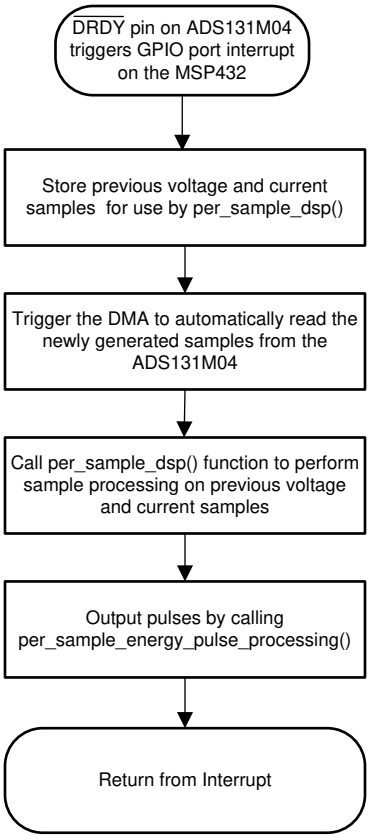

Figure 4-4. Background Process

In the background process, the previously-obtained voltage samples ($V_{phx}[N – 2]$) and previously obtained current samples ($I_{phx}[N – 2]$) are stored so that they can be used later by the per_sample_dsp function, which is responsible for updating the intermediate dot product quantities used to calculate metrology parameters. After the previously-obtained voltage and current samples are stored, communication to the ADS131M08 is enabled by asserting the chip select signal low. The DMA is then configured to both send a request for the newest current and voltage samples ($I_{phx}[N – 1]$ and $V_{phx}[N – 1]$) of the ADS131M08 device and also to receive the data packet response from the ADS131M08. The request and reception of the current samples is done automatically by the DMA module instead of it being done by the software.

The *ADS131M08 ADC Sample Request Packet* figure shows the packet that is transmitted by the DMA of the MSP432 MCU and the response packet from the ADS131M08 that is received and assembled by the DMA as well. The transmission and reception packets contain ten words, where each word is three bytes long.

MSP432 Transmit	Command[N+1] / Dummy Write (3 bytes =0x000000)	Dummy Write (3 bytes= 0x000000)	Dummy Write (3 bytes= 0x000000)	Dummy Write (3 bytes= 0x000000)	Dummy Write (3 bytes= 0x000000)	Dummy Write (3 bytes= 0x000000)	Dummy Write (3 bytes= 0x000000)	Dummy Write (3 bytes= 0x000000)	Dummy Write (3 bytes= 0x000000)	Dummy Write (3 bytes= 0x000000)
MSP432 Receive	Response to Command[N]/Not used in design (3 bytes)	ADS131M04 Ch 0 Sample (3 bytes, MSB sent first)	ADS131M04 Ch 1 Sample (3 bytes, MSB sent first)	ADS131M04 Ch 2 Sample (3 bytes, MSB sent first)	ADS131M04 Ch 3 Sample (3 bytes, MSB sent first)	ADS131M04 Ch 5 Sample (3 bytes, MSB sent first)	ADS131M04 Ch 6 Sample (3 bytes, MSB sent first)	ADS131M04 Ch 7 Sample (3 bytes, MSB sent first)	ADS131M04 Ch 8 Sample (3 bytes, MSB sent first)	ADS131M04 Packet CRC (3 bytes, MSB sent first)

Figure 4-5. ADS131M08 ADC Sample Request Packet

When requesting the ADC data from the ADS131M08 device, the first word that has to be sent to the ADS131M08 is the command word. Since the test software does not need to change the settings of the ADS131M08 or read any registers during typical ADC sample readouts, a NULL command is sent to the ADS131M08, which allows you to get the ADC samples from the ADS131M08 without changing the state of the device. The actual size of the null command is 16-bits; however, since 24-bit words are used, the 16-bit command must be padded with an extra value of 0x00 at the end of the command. The NULL command word sent therefore has a value of 0x000000. While the MSP432 is shifting out the command word, the MSP432 is simultaneously shifting in the response word to the command word of the previous packet. The response word to a NULL command is the contents of the STATUS register. The contents of the STATUS register is not used in this design so the first word received from the ADS131M08 is ignored.

After writing the command word, it is necessary for a dummy write to be performed for each byte that is to be read. The dummy byte write is necessary to enable the SPI clock, which is necessary to read a byte from the ADS131M08 device. For each dummy byte write, a value of 0x00 is written to the SPI transmit register for EUSCIB0. Immediately after writing the command byte, writing three dummy bytes allows the MSP432 MCU to receive the 3-byte ADC value from channel 0 of the ADS131M08. Writing the next 21 dummy bytes gets the ADC data for channel 1–7. Finally, writing the next three dummy bytes gets the CRC word. The CRC word is 24-bits; however, note that the actual CRC is only 16-bits, which are placed in the most significant bits of the 24-bit word. As a result, when parsing the CRC word, the last byte is not needed (note though that the dummy write for this zero-padded byte must still be sent though for proper ADS131M08 operation).

In parallel to receiving the newest current samples from the ADS131M08 using the DMA, the ADS131M08 is currently sampling the next voltage ($V_{phx}[N]$) and current samples ($I_{phx}[N]$) and the test software also performs per-sample processing on the last voltage ($V_{phx}[N – 2]$) and current samples ($I_{phx}[N – 2]$) obtained from the ADS131M08. This per-sample processing is used to update the intermediate dot product quantities that are used to calculate the metrology parameters. After sample processing, the background process uses the "per_sample_energy_pulse_processing" for the calculation and output of energy-proportional pulses. Once the per_sample_energy_pulse_processing is completed, the test software exits from the port ISR.

4.3.1 per_sample_dsp( )

The *per_sample_dsp Function* figure shows the flowchart for the per_sample_dsp() function. The per_sample_dsp() function is used to calculate intermediate dot product results that are fed into the foreground process for the calculation of metrology readings. Voltage samples, fundamental voltage samples, current samples, active power, reactive power, fundamental active power, and fundamental reactive power, are all accumulated in 64-bit registers.

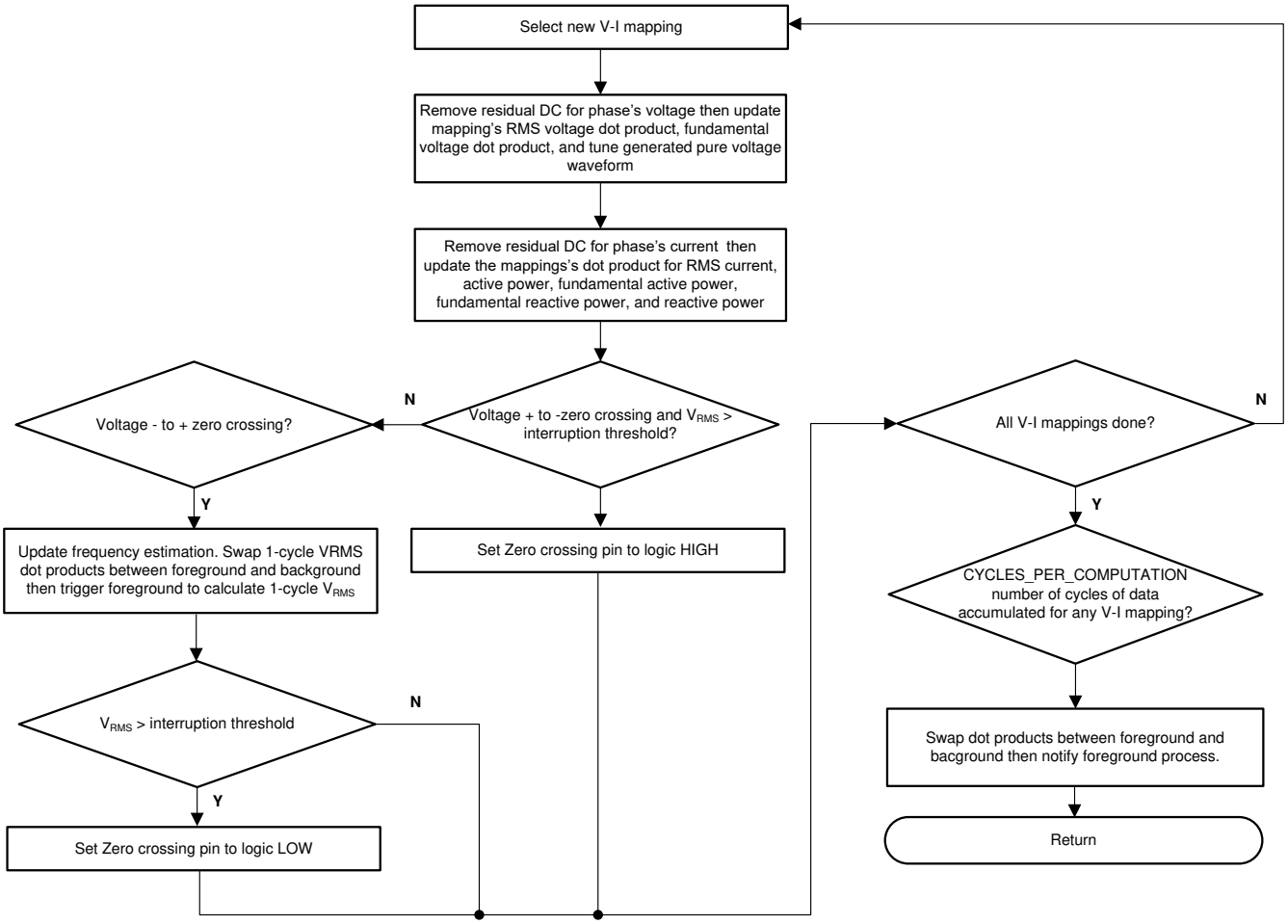

Figure 4-6. per_sample_dsp Function

After CYCLES_PER_COMPUTATION number of cycles (10 cycles if f_{NOM} = 50 and 12 cycles if f_{Nom} = 60) have been accumulated, the background process triggers the foreground function to calculate the final values of RMS voltage and current; active, reactive, and apparent powers; active, reactive, and apparent energy; frequency; power factor; fundamental voltage, fundamental current, fundamental active power, fundamental reactive power, and fundamental apparent power; voltage underdeviation and voltage overdeviation; and Voltage THD and Current THD. In the software, there are two sets of dot products: at any given time, one is used by the foreground for calculation and the other used as the working set by the background. After the background process has sufficient samples, it swaps the two dot products so that the foreground uses the newly acquired dot products that the background process just calculated and the background process uses a new empty set to calculate the next set of dot products.

Whenever there is a leading-edge zero-crossing (− to + voltage transition) on a voltage channel, the per_sample_dsp() function is also responsible for updating the frequency (in samples per cycle) of the corresponding phase and triggering the calculation of the foreground of the 1-cycle V_{RMS} reading. This 1-cycle V_{RMS} reading is a different calculation than the V_{RMS} reading that is updated every CYCLES_PER_COMPUTATION number of cycles. The 1-cycle V_{RMS} reading is specifically used for updating

the sag, swell, and interruption state variables. The 1-cycle V_{RMS} calculation uses the same dot-product swapping scheme as the scheme used for the CYCLES_PER_COMPUTATION dot products.

The per_sample_dsp function is also responsible for outputting a voltage zero crossing pin. Whenever there is a negative to positive zero crossing on a voltage channel and the corresponding 1-cycle V_{RMS} reading of the voltage channel is greater than the interruption threshold, a falling edge is asserted on this pin. If there is a positive to negative zero crossing on a voltage channel and the 1-cycle V_{RMS} reading of the voltage channel is greater than the interruption threshold, a rising edge is asserted on this pin. To reduce the impact of outputting the zero crossing pin on the accuracy of the design, the zero crossing output should not be selected to be on a GPIO pin connected to a LED.

The following sections describe the various elements of electricity measurement in the per_sample_dsp function.

4.3.1.1 Voltage and Current ADC Samples

The raw ADS131M04 samples are signed integers and any stray DC or offset value on these converters are removed using a DC tracking filter. A separate DC estimate for all voltages and currents is obtained using the filter, voltage, and current samples, respectively. This estimate is then subtracted from each voltage and current sample.

The resulting instantaneous voltage and current samples are used to generate the following intermediate results:

- Accumulated squared values of voltage and currents, which is used for V_{RMS} and I_{RMS} calculations, respectively
- Accumulated energy samples to calculate active energy
- Accumulated energy samples using current and 90° phase-shifted voltage to calculate reactive energy

The foreground process processes these accumulated values.

4.3.1.2 Pure Waveform Samples

To calculate the fundamental and THD readings, the software generates a pure sinusoid waveform and locks it to the fundamental of the incoming voltage waveform. Since the generated waveform is locked to the fundamental of the incoming voltage, the correlation of this pure waveform with the waveform from the voltage ADC can be used to find the amplitude of the fundamental component of the waveform sensed by the voltage ADC. Similarly, the correlation of the current and the pure voltage waveform can be used to calculate the fundamental active power. For fundamental reactive power, the correlation of the 90° shifted pure waveform and the current can be used for calculating this parameter.

To generate a sine wave, information on the amplitude, phase, and frequency of the desired waveform is necessary. For the generated pure waveform, the amplitude is set to full scale to maximize the value of the fundamental dot products, the frequency is set to the measured frequency (in units of cycles/sample) that is used to calculate the mains frequency in final real-world units of Hertz, and the phase of the generated waveform is iteratively adjusted so that it is locked to the phase of the fundamental voltage. After the frequency is correctly calculated and the generated phase of the waveform is locked to the fundamental voltage, the fundamental readings can then be correctly calculated.

4.3.1.3 Frequency Measurement and Cycle Tracking

The instantaneous voltage, currents, active powers, and reactive powers are accumulated in 64-bit registers. A cycle tracking counter keeps track of the number of cycles accumulated. When CYCLES_PER_COMPUTATION number of cycles have been accumulated, the background process stores these accumulation registers and notifies the foreground process to produce the average results, such as RMS and power values.

For frequency measurements, a straight line interpolation is used between the zero crossing voltage samples. The *Frequency Measurement* figure the samples near a zero cross and the process of linear interpolation.

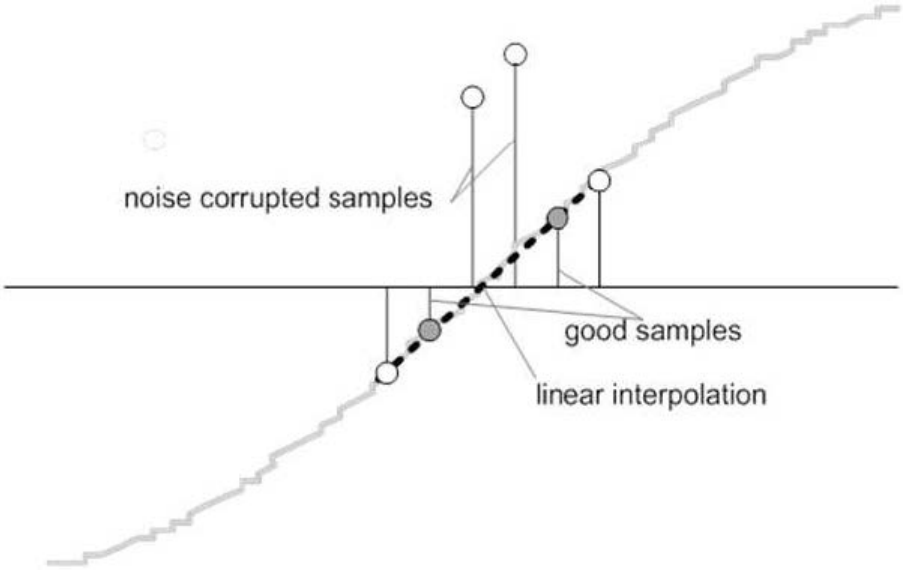

Figure 4-7. Frequency Measurement

Because noise spikes can also cause errors, the application uses a rate-of-change check to filter out the possible erroneous signals and make sure that the two points are interpolated from genuine zero crossing points. For example, with two negative samples, a noise spike can make one of the samples positive, thereby making the negative and positive pair appear as if there is a zero crossing.

The resultant cycle-to-cycle timing goes through a weak low-pass filter to further smooth out any cycle-to-cycle variations. This filtering results in a stable and accurate frequency measurement that is tolerant of noise.

4.3.2 LED Pulse Generation

This implementation supports outputting energy consumption by using pulses that have a frequency that is proportional to the amount of energy consumption. This information can be used to accurately calibrate the system for energy accuracy measurement. Typically, the measuring element (the MSP432 microcontroller) is responsible for generating pulses proportional to the energy consumed. To serve both these tasks efficiently, the pulse generation must be accurate with relatively little jitter. Although time jitters are not an indication of bad accuracy, time jitters give a negative indication of the overall accuracy of the meter. The jitter must be averaged out due to this negative indication of accuracy.

This application uses average power to generate these energy pulses. The average power accumulates at every $\overline{\text{DRDY}}$ port ISR interrupt, thereby spreading the accumulated energy from the previous one-second time frame evenly for each interrupt in the current one-second time frame. This accumulation process is equivalent to converting power to energy. When the accumulated energy crosses a threshold, a pulse is generated. The amount of energy above this threshold is kept and a new energy value is added on top of the threshold in the next interrupt cycle. Because the average power tends to be a stable value, this way of generating energy pulses is very steady and free of jitter.

The threshold determines the energy "tick" specified by equipment manufacturers and is a constant. The tick is usually defined in pulses-per-kWh or just in kWh. One pulse must be generated for every energy tick. For example, in this application, the number of pulses generated per kWh is set to 6400 for active and reactive energies. The energy tick in this case is 1 kWh / 6400. Energy pulses are generated and available on a header. The pulses can be viewed by probing the appropriate pins on the MSP432 LaunchPad. In this design, there are active and reactive pulses for a user-definable V-I mapping.

The *Pulse Generation for Energy Indication* figure shows the flow diagram for pulse generation.

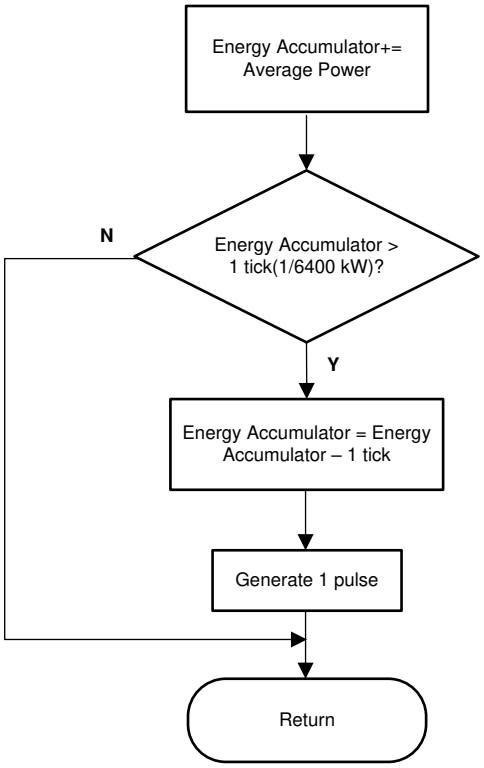

Figure 4-8. Pulse Generation for Energy Indication

The average power is in units of 0.001 W and a 1-kWh threshold is defined as:

1-kWh threshold = 1 / 0.001 × 1 kW × (Number of interrupts per sec) × (Number of seconds in one hour) = 1000000 × 7812.5 × 3600 = 0x19945CA26200

4.3.3 Phase Compensation

The user must compensate the relative phase shift between voltage and current samples to ensure accurate measurements. The implementation of the phase-shift compensation consists of an integer part and a fractional part. The integer part is realized by providing an N samples delay. The fractional part is realized by a one-tap finite impulse response (FIR) filter that interpolates between two samples, similar to the FIR filter used for providing 90°-shifted voltage samples for reactive energy measurements. In the test software, a lookup table provides the filter coefficients that are used to create the fractional delays. The lookup table provides fractional phase shifts as small as 1/256th of a sample. The 7812.5-Hz sample rate used in this application corresponds to a 0.0090° degree resolution at 50 Hz. In addition to the filter coefficients, the lookup table also has an associated gain variable for each set of filter coefficients. This gain variable is used to cancel out the resulting gain from using a certain set of filter coefficients.

An alternative option to the software phase compensation used in this design is to use the phase compensation feature on the ADS131M04 device. If this hardware phase compensation scheme is used, filter coefficients are not necessary so it is not needed to divide by the gain of the filter coefficients.

5 Metrology Accuracy Testing

5.1 Test Setup

To test for metrology accuracy of the design, a source generator is used to provide the 1 voltage and 3 currents to the system. Each of the three currents use the same voltage signal for the corresponding calculation of power. In this design, a nominal voltage of 230 V, calibration currents of 1 A, and nominal frequency of 50 Hz are used. Under these voltage and current conditions, each voltage-current mapping has RMS gain calibration, power gain calibration, and power phase correction performed on them. Once the system has been calibrated, the metrology testing is then performed.

After calibrating the design, different voltage and current condtions are applied to the design. When the voltage and currents are applied to the system, the design outputs active energy pulses for a selected voltage-current mapping at a rate of 6400 pulses/kWh. This pulse output is fed into a reference meter (in the test equipment for this design, this pulse output is integrated in the same equipment used for the source generator) that determines the energy % error for that voltage-current mapping based on the actual energy provided to the system and the measured energy as determined by the active energy output pulse of the design. In addition to the energy error tests, the RMS voltage % error, RMS current % error, and active power % error tests are performed as well. Since the time between pulses at low currents is relatively large, the averaging interval at lower currents would be much larger for the active energy reading compared to the averaging time at lower currents for the active power readings. The longer averaging time of the active energy readings leads to averaging out more noise, which results in better results for the active energy % error readings than the active power % error readings.

For the active energy error, current is varied from 50 mA to 20 A . A phase shift of 0° (unity power factor), 60° (0.5 power factor, inductive), and –60° (0.5 power factor, capacitive) is applied between the voltage and current waveforms fed to the reference design. Based on the error from the active energy output pulse, a plot of active energy % error versus current is created at 0°, 60°, and –60° phase shifts for each voltage-current mapping. When testing the accuracy of a voltage-current mapping, the same currents and phase shift are applied simulatenously for all voltage-current mappings.

To test RMS accuracy, the RMS readings were used from the GUI since the pulse output that was used for the energy accuracy tests cannot be used for RMS voltage and current. For the voltage testing, 1-A current is applied for each phase and the voltage is varied from 9–270 V on each phase simultaneously. After applying each voltage, the resulting RMS voltage reading from the GUI is logged for each phase after the readings stabilize. Once the measured RMS voltage readings are obtained from the GUI, the actual RMS voltage readings are obtained from the reference meter, which is necessary because the source generator may not accurately generate the requested values for voltage, especially at small voltages. With the reference meter measured RMS voltage and the RMS voltage value of the GUI, the RMS voltage % error is calculated. A similar process is used to calculate the RMS current % error and active power % error by using 230 V for each phase and varying current from 100 mA to 20 A.

5.2 Results

This section provides details of the test results.

Table 5-1. Voltage-Current A Mapping Active Energy % Error vs Current

CURRENT (A)	0°	60°	–60°
0.05	0.066	0.016	–0.079
0.1	0.062	0.03	–0.054
0.25	0.036	–0.009	–0.074
0.5	–0.036	–0.023	0.033
1	–0.0003	0.04	–0.0637
2	–0.0105	0.052	–0.059
5	–0.0263	–0.011	–0.064
7.5	–0.017	–0.007	–0.05
10	–0.0273	–0.022	–0.056
12.5	–0.072	–0.048	–0.118
15	–0.0923	–0.075	–0.124
17.5	–0.121	–0.094	–0.136
20	–0.12	–0.117	–0.117

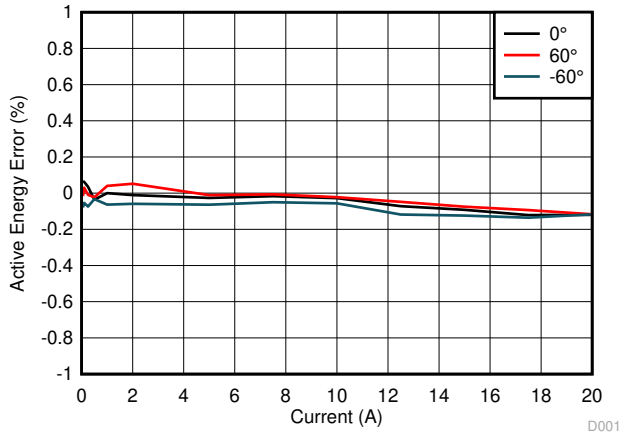

Figure 5-1. Voltage-Current A Mapping Active Energy % Error vs Current

Table 5-2. Voltage-Current B Mapping Active Energy % Error vs Current

CURRENT (A)	0°	60°	–60°
0.05	–0.095	–0.067	0.037
0.1	–0.066	–0.102	–0.054
0.25	–0.083	0	–0.068
0.5	0.028	–0.023	–0.001
1	–0.013	–0.003	–0.03
2	–0.049	–0.014	–0.026
5	–0.014	–0.044	–0.035
7.5	–0.033	–0.064	–0.037
10	–0.043	–0.084	–0.031
12.5	–0.1	–0.107	–0.103
15	–0.133	–0.107	–0.108
17.5	–0.151	–0.137	–0.129
20	–0.13	–0.173	–0.125

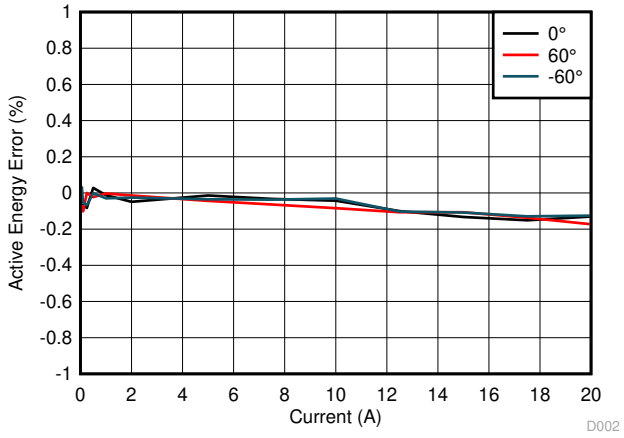

Figure 5-2. Voltage-Current B Mapping Active Energy % Error vs Current

Table 5-3. Voltage-Current C Mapping Active Energy % Error vs Current

CURRENT (A)	0°	60°	–60°
0.05	0.061	0.049	0.112
0.1	–0.02	0.139	–0.062
0.25	–0.024	0.058	0.111
0.5	–0.03	0.039	–0.065
1	–0.016	–0.022	0
2	–0.0243	0.027	–0.08
5	–0.0257	0.032	–0.035
7.5	–0.022	–0.01	–0.067
10	–0.0623	0.011	–0.062
12.5	–0.0823	–0.023	–0.155
15	–0.1	–0.057	–0.135
17.5	–0.153	–0.11	–0.151
20	–0.153	–0.127	–0.207

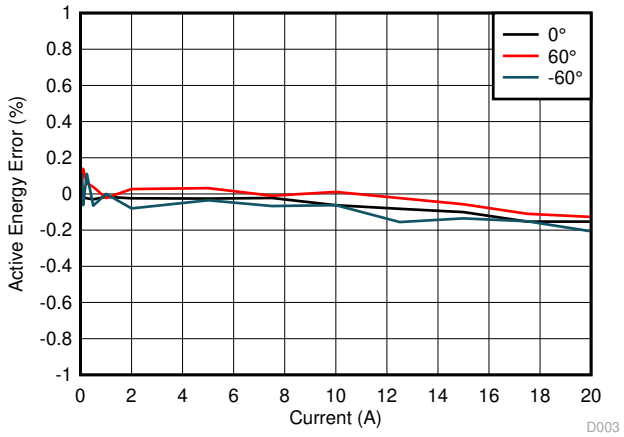

Figure 5-3. Voltage-Current C Mapping Active Energy % Error vs Current

Table 5-4. RMS Current A % Error

CURRENT (A)	GUI CURRENT READING (A)	REFERENCE METER READING (A)	% ERROR
0.1	0.101219	0.10005	1.1684
0.25	0.25053	0.25001	0.2080
0.5	0.5002	0.50002	0.0360
1	1.00053	1.0001	0.0430
2	2.0008	2.0001	0.0350
5	4.99939	4.9996	–0.0042
7.5	7.501	7.4991	0.0253
10	9.99727	10.001	–0.0373
12.5	12.5031	12.507	–0.0312
15	14.9977	15.003	–0.0353
17.5	17.4938	17.501	–0.0411
20	19.9816	20	–0.0920

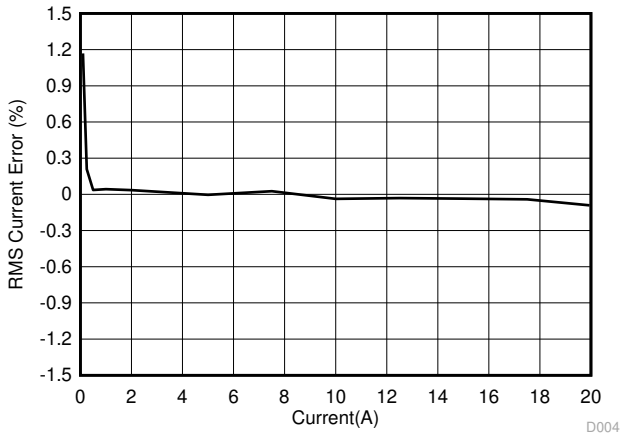

Figure 5-4. RMS Current A % Error

Table 5-5. RMS Current B % Error

CURRENT (A)	GUI CURRENT READING (A)	REFERENCE METER READING (A)	% ERROR
0.1	0.10118	0.099974	1.2063
0.25	0.250601	0.24996	0.2564
0.5	0.500613	0.4999	0.1426
1	1.00036	1.0001	0.0260
2	2.00081	2.002	–0.0594
5	5.00075	4.9988	0.0390
7.5	7.49846	7.4982	0.0035
10	9.99771	9.9977	0.0001
12.5	12.4902	12.504	–0.1104
15	14.9914	15	–0.0573
17.5	17.483	17.502	–0.1086
20	19.9802	20.002	–0.1090

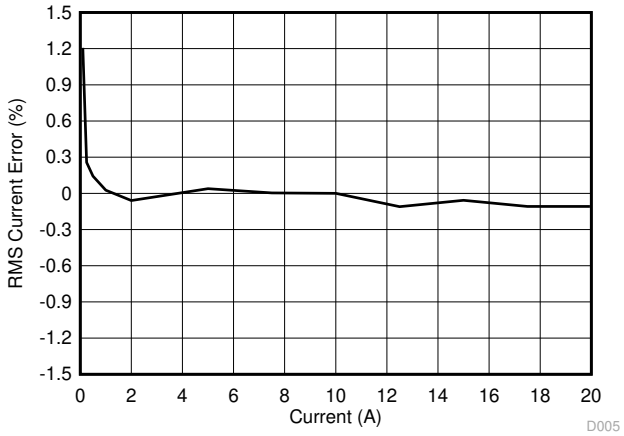

Figure 5-5. RMS Current B % Error

Table 5-6. RMS Current C % Error

CURRENT (A)	GUI CURRENT READING (A)	REFERENCE METER READING (A)	% ERROR
0.1	0.101275	0.1	1.2750
0.25	0.25027	0.24996	0.1240
0.5	0.499771	0.4999	–0.0258
1	0.999515	0.99991	–0.0395
2	1.99821	2	–0.0895
5	4.99569	4.9989	–0.0642
7.5	7.49579	7.4988	–0.0401
10	9.98942	10	–0.0938
12.5	12.4857	12.504	–0.1464
15	14.9768	15	–0.1547
17.5	17.4664	17.499	–0.1863
20	19.9472	19.999	–0.2590

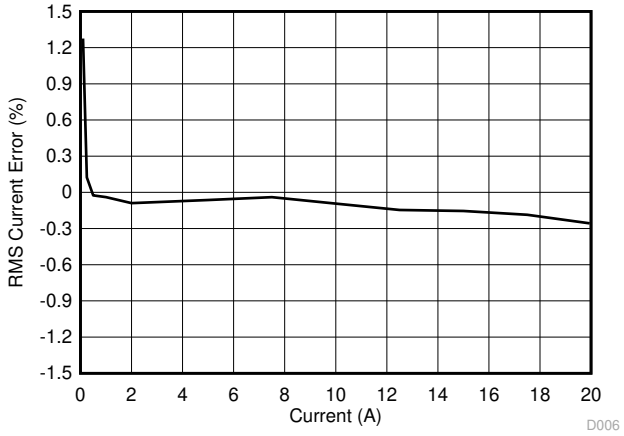

Figure 5-6. RMS Current C % Error

Table 5-7. RMS Voltage % Error

VOLTAGE (V)	GUI VOLTAGE READING (V)	REFERENCE METER READING (V)	% ERROR
9	9.001	9.005	–0.0444
10	10	10.004	–0.0400
30	29.996	30.003	–0.0233
50	49.991	50.004	–0.0260
70	69.995	70.012	–0.0243
100	99.983	100.01	–0.0270
120	119.982	120.02	–0.0317
140	139.99	140.03	–0.0286
160	159.977	160.01	–0.0206
180	179.992	180.03	–0.0211
200	199.98	200.03	–0.0250
220	219.978	220.05	–0.0327
230	229.971	230.05	–0.0343
240	239.993	240.06	–0.0279
270	270	270.08	–0.0296

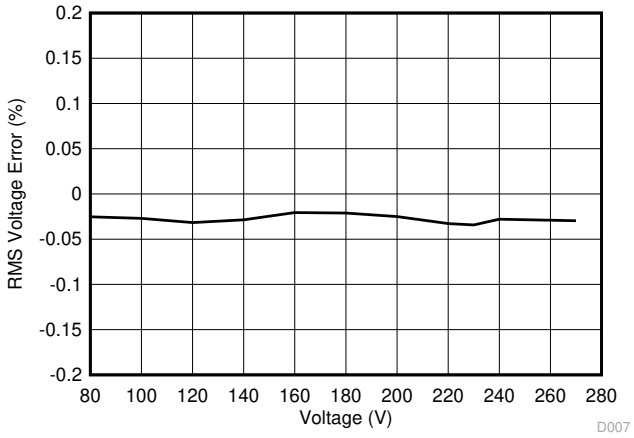

Figure 5-7. RMS Voltage % Error

Table 5-8. Voltage-Current A Mapping Active Power % Error vs Current, 0° Phase Shift

CURRENT (A)	GUI ACTIVE POWER READING (W)	REFERENCE METER READING (W)	% ERROR
0.1	23.046	23.016	0.1303
0.25	57.489	57.515	–0.0452
0.5	115.053	115.02	0.0287
1	230.021	230.07	–0.0213
2	460.111	460.12	–0.0020
5	1149.75	1150.2	–0.0391
7.5	1725.35	1725.2	0.0087
10	2299.41	2300.7	–0.0561
12.5	2875.52	2877	–0.0514
15	3449.55	3451.8	–0.0652
17.5	4021.51	4026.2	–0.1165
20	4596	4601	–0.1087

Table 5-9. Voltage-Current A Mapping Active Power % Error vs Current, 60° Voltage-Current B Mapping Active Power % Error vs Current, 0° Phase Shift

CURRENT (A)	GUI ACTIVE POWER READING (W)	REFERENCE METER READING (W)	% ERROR
0.1	11.46	11.504	–0.3825
0.25	28.783	28.744	0.1357
0.5	57.539	57.491	0.0835
1	115.137	115.05	0.0756
2	230.057	229.95	0.0465
5	574.89	574.84	0.0087
7.5	862.212	862.39	–0.0206
10	1150.5	1150.3	0.0174
12.5	1436.45	1437.7	–0.0869
15	1724.55	1725	–0.0261
17.5	2010.12	2012	–0.0934
20	2297.69	2299.4	–0.0744

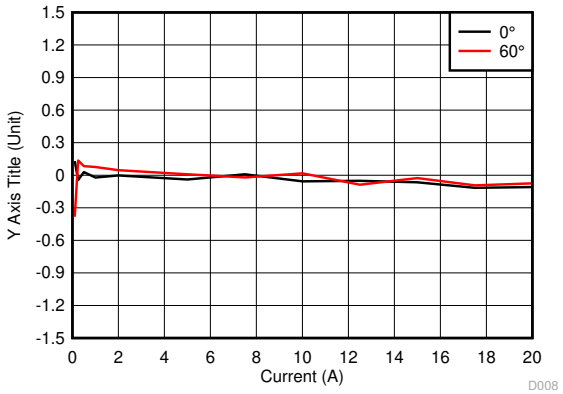

Figure 5-8. Voltage-Current A Mapping Active Power % Error vs Current 90

Table 5-10. Voltage-Current B Mapping Active Power % Error vs Current, 0° Phase Shift

CURRENT (A)	GUI ACTIVE POWER READING (W)	REFERENCE METER READING (W)	% ERROR
0.1	23.023	23.002	0.0913
0.25	57.48	57.501	–0.0365
0.5	115.07	114.99	0.0696
1	230.017	229.99	0.0117
2	460.1928	460	0.0419
5	1149.66	1149.8	–0.0122
7.5	1724.5	1725	–0.0290
10	2299.43	2299.9	–0.0204
12.5	2873.59	2875.1	–0.0525
15	3446.3	3450.6	–0.1246
17.5	4020.2	4026.5	–0.1565
20	4595	4600	–0.1087

Table 5-11. Voltage-Current B Mapping Active Power % Error vs Current, 60° Phase Shift

CURRENT (A)	GUI ACTIVE POWER READING (W)	REFERENCE METER READING (W)	% ERROR
0.1	11.547	11.501	0.4000
0.25	28.76	28.745	0.0522
0.5	57.594	57.501	0.1617
1	114.998	115.01	–0.0104
2	230.049	229.94	0.0474
5	574.649	574.99	–0.0593
7.5	862.059	862.46	–0.0465
10	1149.61	1150	–0.0339
12.5	1435.72	1436.9	–0.0821
15	1722	1724.2	–0.1276
17.5	2008.68	2010.9	–0.1104
20	2295.41	2299.2	–0.1648

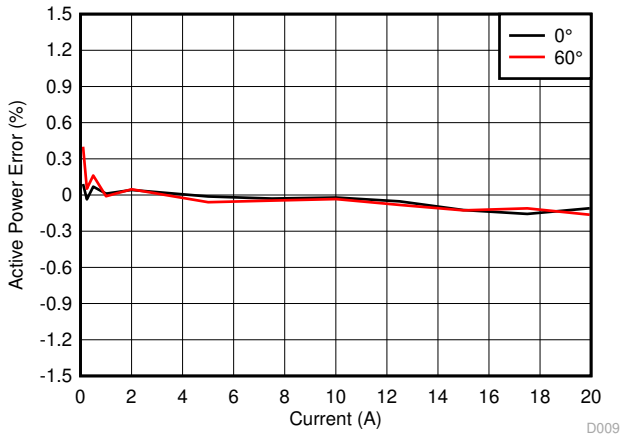

Figure 5-9. Voltage-Current B Mapping Active Power % Error vs Current

Table 5-12. Voltage-Current C Mapping Active Power % Error vs Current,0° Phase Shift

CURRENT (A)	GUI ACTIVE POWER READING (W)	REFERENCE METER READING (W)	% ERROR
0.1	22.999	23.005	–0.0261
0.25	57.5	57.499	0.0017
0.5	115	114.99	0.0087
1	229.914	230.02	–0.0461
2	459.885	460.1	–0.0467
5	1150	1149.52	0.0418
7.5	1724.74	1725.1	–0.0209
10	2299.2	2300.3	–0.0478
12.5	2873.22	2876.2	–0.1036
15	3447.22	3450.8	–0.1037
17.5	4020.78	4025.5	–0.1173
20	4590.42	4600.4	–0.2169

Table 5-13. Voltage-Current C Mapping Active Power % Error vs Current, 60° Phase Shift

CURRENT (A)	GUI ACTIVE POWER READING (W)	REFERENCE METER READING (W)	% ERROR
0.1	11.469	11.498	–0.2522
0.25	28.729	28.687	0.1464
0.5	57.47	57.458	0.0209
1	114.922	114.99	–0.0591
2	229.8	229.885	–0.0370
5	574.507	574.61	–0.0179
7.5	862.051	862.14	–0.0103
10	1149.32	1149.8	–0.0417
12.5	1434.67	1436.2	–0.1065
15	1721.22	1723.1	–0.1091
17.5	2007.75	2010.3	–0.1268
20	2292.83	2297.1	–0.1859

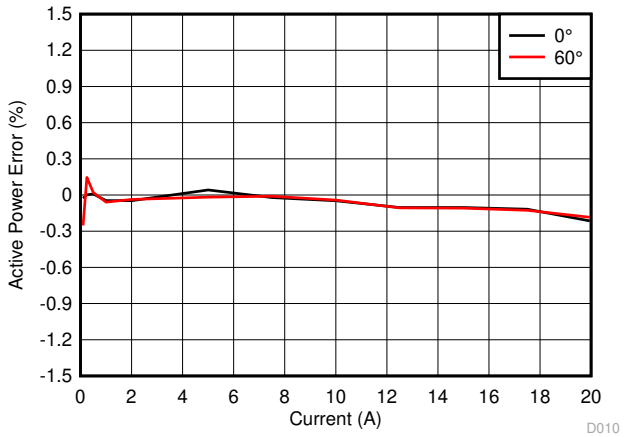

Figure 5-10. Voltage-Current C Mapping Active Power % Error vs Current

6 Schematics

Figure 6-1 through Figure 6-3 illustrate the schematic images.

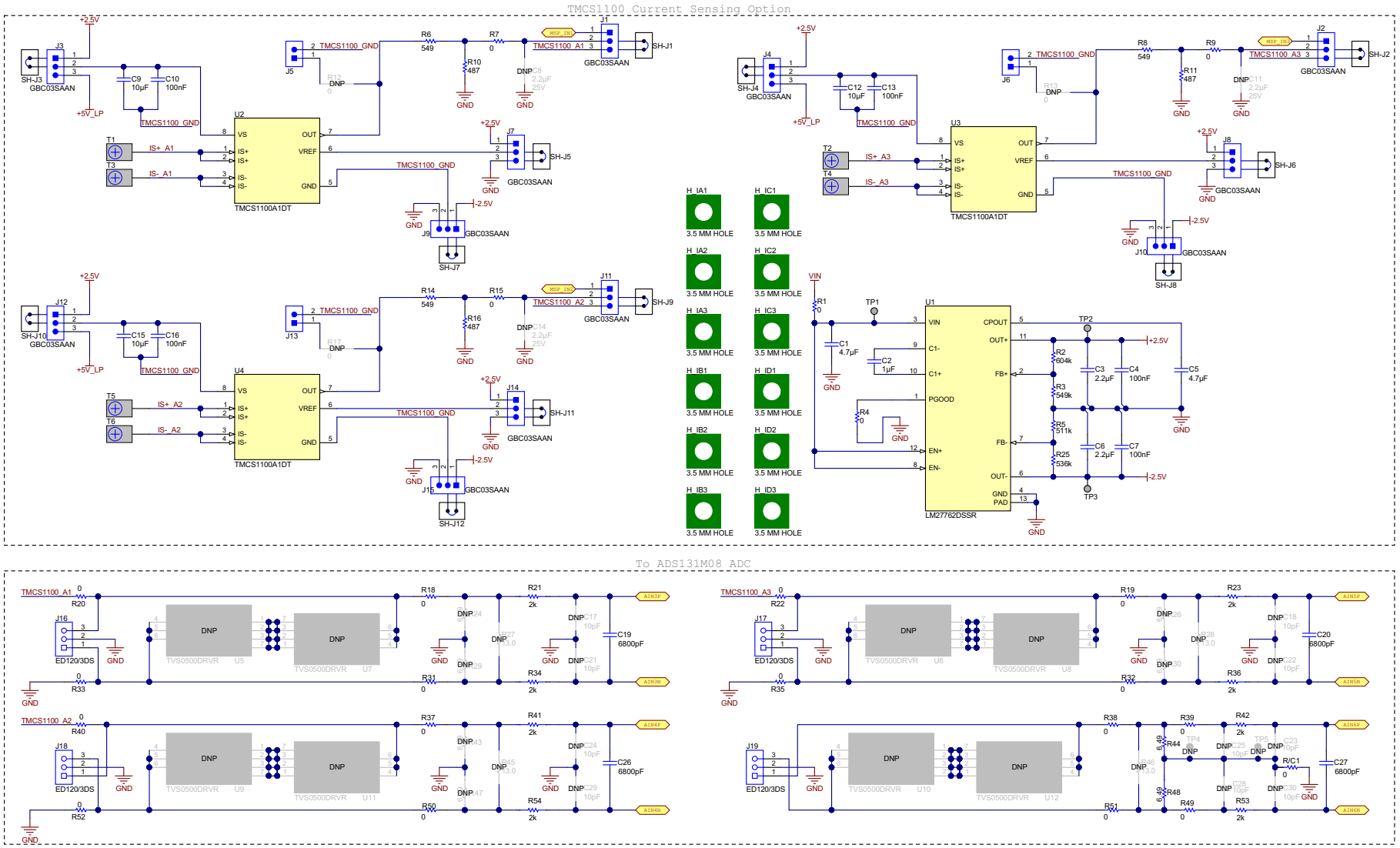

Figure 6-1. Schematic Page 1

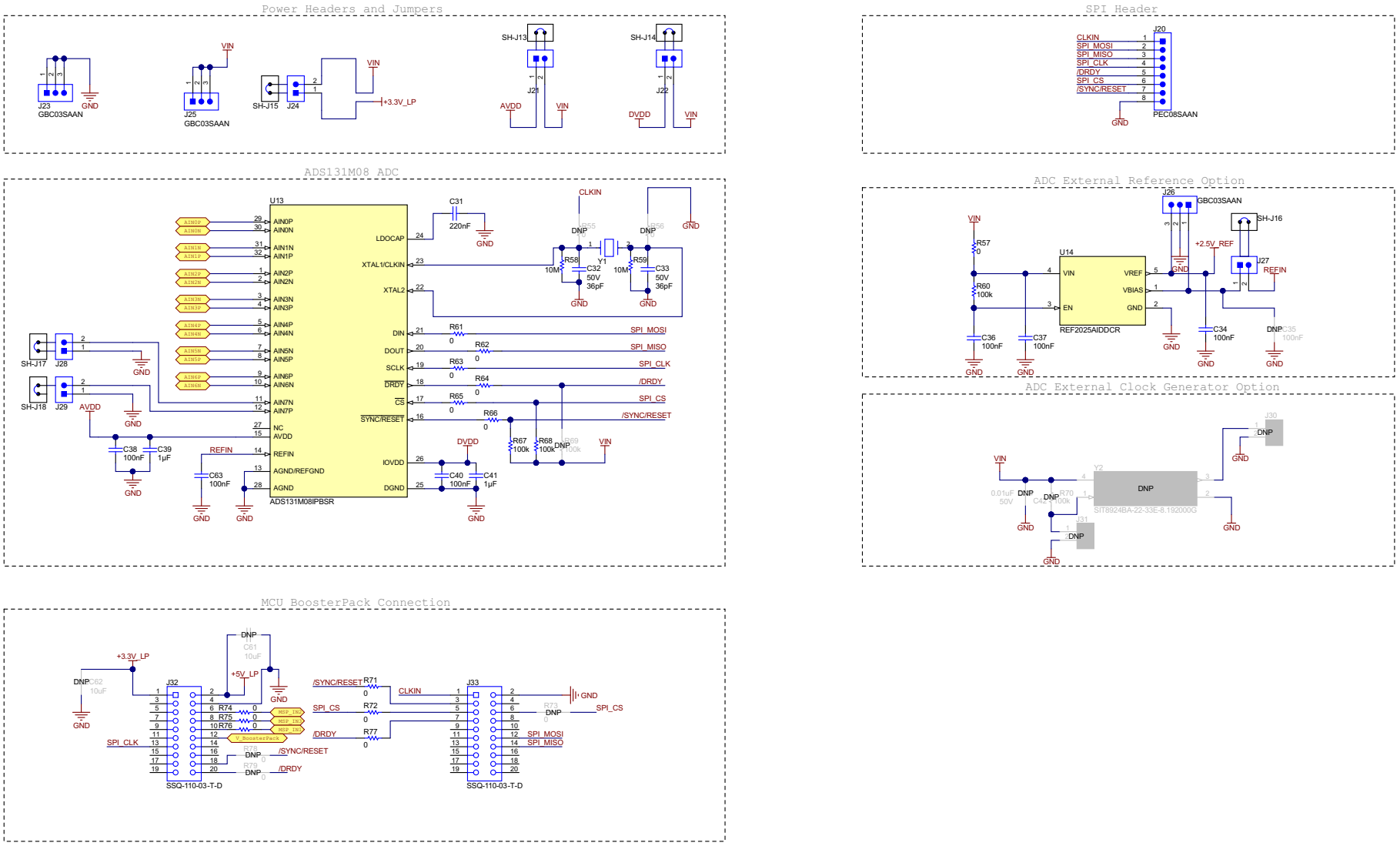

Figure 6-2. Schematic Page 2

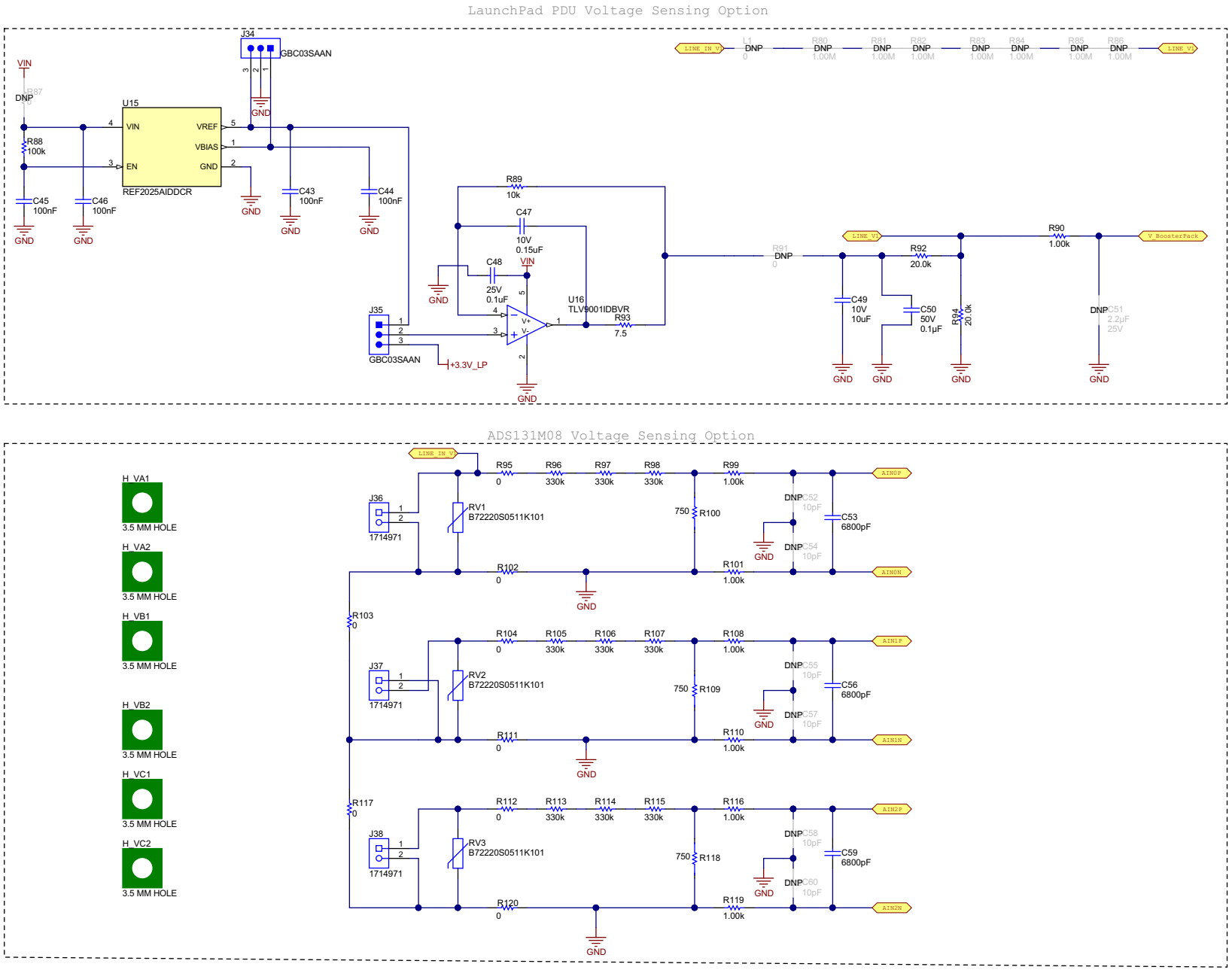

Figure 6-3. Schematic Page 3

7 References

- Texas Instruments, *ADS131M08 8-Channel, Simultaneously-Sampling, 24-Bit, Delta-Sigma ADC Data Sheet*
- Texas Instruments, *TMCS1100 1% High-Precision, Basic Isolation Hall-Effect Current Sensor With ±600-V Working Voltage Data Sheet*
- Texas Instruments, *LM27762 Low-Noise Positive and Negative Output Integrated Charge Pump Plus LDO Data Sheet*
- Texas Instruments, *Isolated RS-232 to UART Converter Reference Design*
- Texas Instruments, *SimpleLink™ MSP432P4111 high-precision ADC MCU LaunchPad™ Development Kit*

IMPORTANT NOTICE AND DISCLAIMER

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

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

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

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

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

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