*Design Guide: TIDA-010062*

# *1-kW, 80+ Titanium, GaN CCM Totem Pole Bridgeless PFC and Half-Bridge LLC With LFU Reference Design*

## Description

This reference design is a digitally controlled, compact 1-kW AC/DC power supply design for server power supply unit (PSU) and telecom rectifier applications. The highly-efficient design supports two main power stages, including a front-end continuous conduction mode (CCM) totem-pole bridgeless power factor correction (PFC) stage. The PFC stage features TI's LMG341x gallium nitride (GaN) FET with integrated driver to provide enhanced efficiency across a wide load range and meet 80 Plus Titanium requirements. The design also supports LMG3422 GaN FET half-bridge inductor-inductor-capacitor (LLC) isolated DC/DC stage to achieve a +12-V DC output at 1 kW. Two control cards use C2000™ Piccolo™ microcontrollers to control both power stages.

## Resources

| | |
|---|---|
| TIDA-010062 | Design Folder |
| TMS320F280049, TMS320F280025 | Product Folder |
| TMS320F280039, TMDSFSIADAPEVM | Product Folder |
| LMG3410, TMCS1100, LMG3422 | Product Folder |
| ISO7721, INA180, ISO7742, OPA2376 | Product Folder |
| TPS560430, UCC27524, UCC28911 | Product Folder |

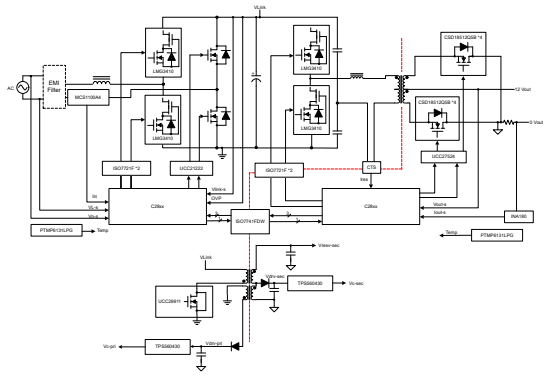
Ask our TI E2E™ support experts

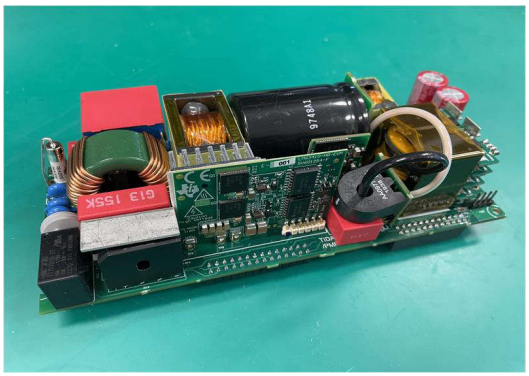
## Features

- 80 Plus Titanium efficiency, η = > 95% at 20%–100% load
- CCM GaN-based totem-pole bridgeless PFC stage with > 98.8% peak efficiency, enabled by LMG341x GaN FET with integrated driver
- Half-bridge LMG342x GaN FET LLC stage with > 98.3% peak efficiency
- Fast load transient, $V_o$ change within 300 mV at 2.5-A / µs slew rate
- Low iTHD, < 10% at 10% load, < 5% at 20% load, < 2% at > 50% load
- Power density 41 W / $in^3$, 38 mm × 66 mm × 160 mm
- Fast serial interface (FSI) based PFC, DC/DC communication
- Live Firmware Update (LFU) on Half-Bridge LLC (F28003x)

## Applications

- Merchant network and server PSU
- Merchant telecom rectifiers
- Industrial AC/DC





# 1 System Description

In network, server, and datacenter systems, high efficiency across a full load range of power supply units (PSU) is a key requirement. Because server computers are continuously working with high power consumption for calculation and communication, high-efficiency PSUs reduce the operating investment. 80 PLUS® is a voluntary certification program intended to promote efficient energy use in computer PSUs. The program certifies products that have greater than 80% energy efficiency at 20%, 50%, and 100% of rated load. The highest level has reached Titanium level.

This reference design is a 1000-W AC/DC power supply for merchant server PSU applications, demonstrating high efficiency operation in a small form factor (160 mm × 66 mm × 38 mm). This reference design consists of a continuous conduction mode (CCM) totem pole power factor correction (PFC) as the front stage, an isolated half bridge LLC as the output stage, and a primary side regulation (PSR) flyback generates bias power for the control stage. The PFC stage operates from an input voltage range of 100-V - 265-V AC RMS and generates a 380-V DC bus. The second stage is made up of an isolated half bridge LLC stage, which generates a 12-V, 84-A nominal output.

## 1.1 Key System Specifications

**Table 1-1. Key System Specifications**

| PARAMETER | TEST CONDITION | MINIMUM | NOMINAL | MAXIMUM | UNIT |
|---|---|---|---|---|---|
| AC input voltage | Low line | 100 | 115 | 132 | V |
| AC input voltage | High line | 180 | 230 | 264 | V |
| Line frequency | | 47 | | 63 | Hz |
| Output power | High line | | 1000 | | Watt |
| Output power | Low line | | 1000 | | Watt |
| PF | 230 $V_{in}$, 100% load | | 0.99 | | |
| iTHD | 10% load | | | <10% | |
| | 100% load | | | <2% | |
| Ripple and noise | 12-V main output | | | ±120 | mV |
| AC holdup time | 100% load, within $V_{out}$ regulation | 8 | 10 | | mS |
| Output voltage regulation | All line and load conditions (±5%) | 11.4 | 12 | 12.6 | V |
| Dynamic loading/transient response | 50% of $I_{out-max}$, 0.5-A/uS, 3300-uF capacitive load | | ±5% | | % |
| Operating ambient | | 0 | 25 | 50 | °C |
| OCP of 12-V output | Shut-down, latch-off | 110 | | 130 | % of $I_{omax}$ |
| OVP of 12-V output | Shut-down, latch-off | 13 | | 14.5 | V |
| OTP | Auto restart | | | | |
| Dimension (mm) | Without shell and golden finger | | 160 × 66 × 38 | | mm |
| Power density | | | 41 | | $W/in^3$ |

# 2 System Overview

## 2.1 Block Diagram

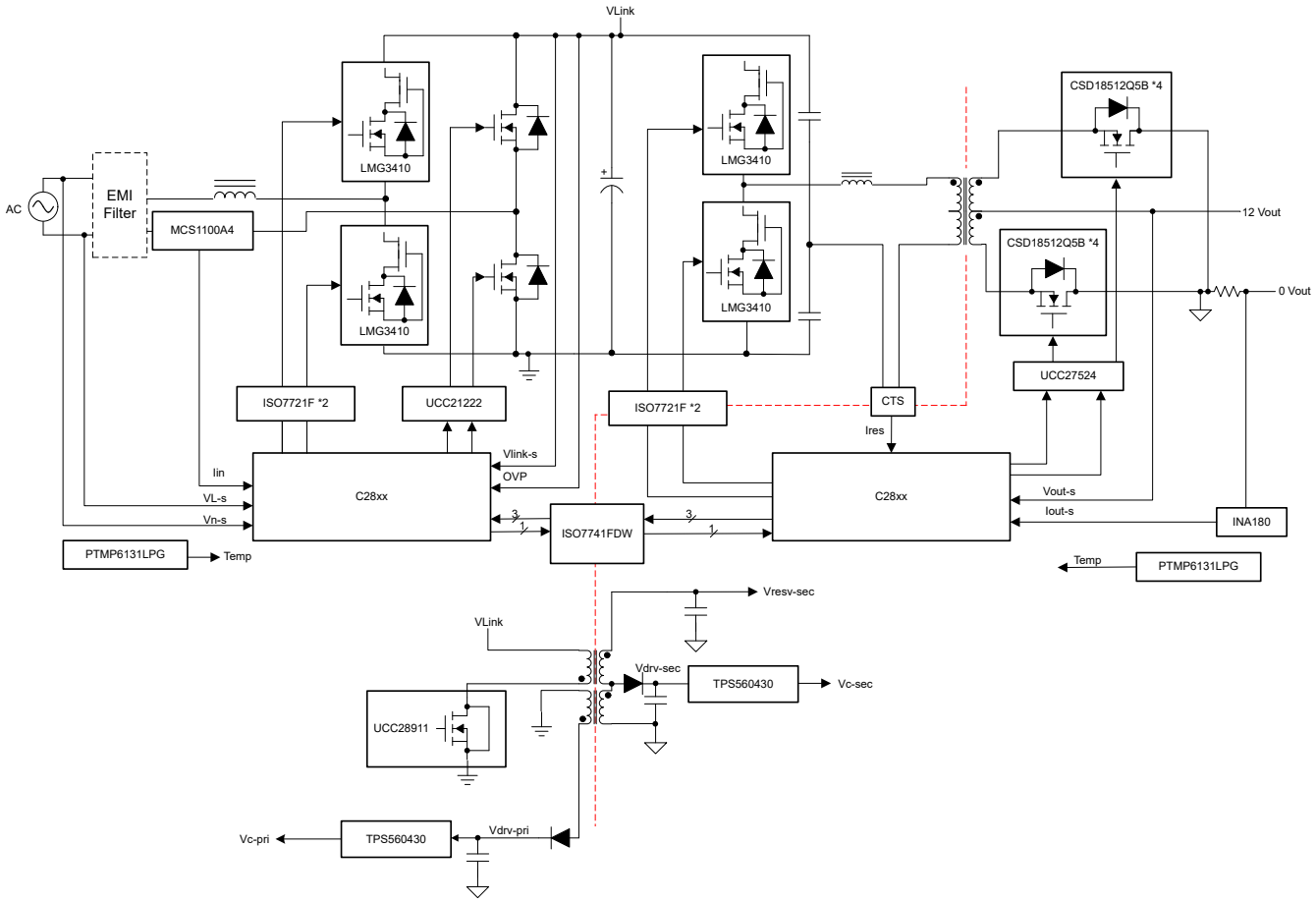



**Figure 2-1. TIDA-010062 Block Diagram**

Figure 2-1 shows the high-level block diagram of the design. In this design, two TMS320F280039, TMS320F280049, or TMS320F280025 C2000™ Piccolo™ control cards are used, respectively, for the controller of the PFC stage and LLC stage, and an LMG3410 GaN daughter card is used for totem pole half-bridge switching. A silicon MOSFET-based synchronous rectifier, driven by the UCC27714 half-bridge driver, reduces power loss in the low-frequency half bridge at the input. A LMG3422 GaN FET synchronous rectifier is driven by the ISO7721 isolator for LLC half-bridge switching. The input current is sensed by a Hall-effect sensor (TMCS1100), and the DC link voltage is processed by a variable-gain amplifier stage built using the OPA237 amplifier for better accuracy at low current levels. The output current is sensed by the INA180 current-sense amplifier. For communication between the primary and secondary control card, Fast Serial Interface (FSI) is sensed by the four-channel digital isolator ISO7742. An onboard auxiliary power supply is implemented using the UCC28911 and provides a 12-V bias for the primary side and a 6-V bias for the secondary side. The TPS560430 is used in both the primary and secondary side for regulated $V_{CC}$ in control cards and control circuits.

## 2.2 Design Considerations

In the totem pole PFC stage, which benefits from gallium nitride (GaN), the switching loss and reverse recovery loss are reduced significantly. CCM control can be implemented with high efficiency. In the LLC stage, which benefits from a high-resolution PWM control of C2000 and powerful calculation ability, both the primary side and secondary side can achieve soft switching with accurate deadtime and SR control.

For higher efficiency, the LLC stage of this design uses an interleaved structure in the transformer to reduce the proximity effect and uses a PCB winding with a copper strip in parallel to reduce the winding loss on the secondary side. In the LLC transformer, there are four secondary PCB windings with an SR MOSFET and a

dual-channel driver, connected in parallel, and three primary windings connected in serial. These windings are not connection joints, but all three use the same wire twisting. For the detailed design, refer to the documents from ASIAINFO®, the transformer vendor for this design.

## 2.3 Highlighted Products

This reference design uses the following highlighted products. Key features for selecting the devices for this reference design are explained. Complete details of the highlighted devices is referred to in the respective product data sheets.

### 2.3.1 LMG3422R050 — 600-V GaN With Integrated Driver and Protection

The LMG342xR050 GaN FET with integrated driver and protection enables designers to achieve new levels of power density and efficiency in power electronics systems. The LMG342xR050 integrates a silicon driver that enables switching speed up to 150 V/ns. TI's integrated precision gate bias results in higher switching SOA compared to discrete silicon gate drivers. This integration, combined with TI's low inductance package, delivers clean switching and minimal ringing in hard-switching power supply topologies. Adjustable gate drive strength allows control of the slew rate from 20 V / ns to 150 V / ns, which can be used to actively control EMI and optimize switching performance. The LMG3425R050 includes ideal diode mode, which reduces third quadrant losses by enabling adaptive dead-time control. Advanced power management features include digital temperature reporting and fault detection. The temperature of the GaN FET is reported through a variable duty cycle PWM output, which simplifies managing device loading. Faults reported include overtemperature, overcurrent, and UVLO monitoring.

### 2.3.2 TMCS1100 — Precision Isolated Current Sense Monitor

The TMCS1100 is an isolated bidirectional Hall-effect current sensor, with high accuracy, excellent linearity, and stability across temperature.

A low-drift, temperature-compensated signal chain provides <1% full-scale error across the entire device temperature range. The output voltage is proportional to input current with four sensitivity options: 50 mV/A, 100 mV/A, 200 mV/A, and 400 mV/A.

The TMCS1100 operates from a single 3-V to 5.5-V power supply, drawing a maximum supply current of 5 mA. All versions are specified over an extended operating temperature range of 40°C to +125°C and are offered in an 8-pin SOIC package.

### 2.3.3 UCC27524 — Dual, 5-A, High-Speed Low-Side Power MOSFET Driver

The UCC2752x family of devices are dual-channel, high-speed, low-side gate-driver devices capable of effectively driving MOSFET and IGBT power switches. Using a design that inherently minimizes shoot-through current, UCC2752x can deliver high peak current pulses of up to 5-A source and 5-A sink into capacitive loads along with rail-to-rail drive capability and extremely small propagation delay (typically 13 ns). In addition, the drivers feature matched internal propagation delays between the two channels. These delays are very well suited for applications requiring dual-gate drives with critical timing, such as synchronous rectifiers. This also enables connecting two channels in parallel to effectively increase current-drive capability or driving two switches in parallel with one input signal. The input pin thresholds are based on TTL and CMOS compatible low-voltage logic, which is fixed and independent of the VDD supply voltage. Wide hysteresis between the high and low thresholds offers excellent noise immunity.

### 2.3.4 UCC27714 — 620-V, 1.8-A, 2.8-A High-Side Low-Side Gate Driver

The UCC27714 is a 600-V high-side, low-side gate driver with 4-A source and 4-A sink current capability that is targeted to drive power MOSFETs or IGBTs. The device comprises of one ground-referenced channel (LO) and one floating channel (HO), which is designed for operating with bootstrap supplies. The device features excellent robustness and noise immunity with capability to maintain operational logic at negative voltages of up to -8 VDC on HS pin (at VDD = 12 V).

### 2.3.5 ISO7721 — High Speed, Robust EMC, Reinforced and Basic Dual-Channel Digital Isolator

The ISO7721x devices are high-performance, dual-channel digital isolators with 3000-$V_{RMS}$ (D package) isolation ratings per UL 1577. The ISO7721x devices provide high electromagnetic immunity and low emissions

at low power consumption, while isolating CMOS or LVCMOS digital I/Os. Each isolation channel has a logic input and output buffer separated by a double capacitive silicon dioxide (SiO2) insulation barrier. The ISO7721x device has both channels in the opposite direction. In the event of input power or signal loss, the default output is high for devices without suffix F and low for devices with suffix F.

### 2.3.6 ISO7740 and ISO7720 — High-Speed, Low-Power, Robust EMC Digital Isolators

The ISO774x devices are high-performance, quad channel digital isolators with 5000 $V_{RMS}$ (DW package) and 3000 $V_{RMS}$ (DBQ package) isolation ratings per UL1577. This family of devices has reinforced insulation ratings according to VDE, CSA, TUV and CQC. The ISO774x devices provide high electromagnetic immunity and low emissions at low power consumption, while isolating CMOS or LV CMOS digital I/Os. Each isolation channel has a logic input and output buffer separated by a double capacitive silicon diodes (SiO2 ) insulation barrier. This device comes with enable pins which can be used to put the respective outputs in high impedance for multi-master driving applications and to reduce power consumption. The ISO7740 device has all four channels in the same direction, the ISO7741 device has three forward and one reverse-direction channels, and the ISO7742 device has two forward and two reverse-direction channels. If the input power or signal is lost, default output is high for devices without suffix F and low for devices with suffix F.

### 2.3.7 OPA237 — Single-Supply Operational Amplifier

The OPA237 operational amplifier (op amp) family is one of TI's micro amplifiers in a series of miniature products. In addition to small size, these devices feature low offset voltage, low quiescent current, low bias current, and a wide supply range. They are ideal for single-supply applications. When operated from a single supply, the input common-mode range extends below ground, and the output can swing to within 10 mV of ground.

### 2.3.8 INAx180 — Low- and High-Side Voltage Output, Current-Sense Amplifiers

The INA180, INA2180, and INA4180 (INAx180) current sense amplifiers are designed for cost optimized applications. These devices are part of a family of current-sense amplifiers (also called current shunt monitors) that sense voltage drops across current-sense resistors at common-mode voltages from –0.2 V to +26 V, independent of the supply voltage. The INAx180 integrate a matched resistor gain network in four, fixed-gain device options: 20 V/V, 50 V/V, 100 V/V, or 200 V/V. This matched gain resistor network minimizes gain error and reduces the temperature drift.

All of these devices operate from a single 2.7-V to 5.5-V power supply. The single-channel INA180 draws a maximum supply current of 260 µA; whereas, the dual-channel INA2180 draws a maximum supply current of 500 µA, and the quad channel draws a maximum supply current of 900 µA.

The INA180 is available in a 5-pin, SOT-23 package with two different pin configurations. The INA2180 is available in an 8-pin, VSSOP package. The INA4180 is available in a 14-pin, TSSOP package. All device options are specified over the extended operating temperature range of –40°C to +125°C.

### 2.3.9 TPS560430 — SIMPLE SWITCHER® 4-V to 36-V, 600-mA Synchronous Step-Down Converter

The TPS560430 is an easy-to-use synchronous step-down DC-DC converter capable of driving up to 600-mA load current. With a wide input range of 4 V to 36 V, the device is suitable for a wide range of applications from industrial to automotive for power conditioning from an unregulated source.

The TPS560430 has 1.1-MHz and 2.1-MHz operating frequency versions for either high efficiency or small solution size. The TPS560430 also has a FPWM (forced PWM) version to achieve constant frequency and small output voltage ripple over the full load range. Soft-start and compensation circuits are implemented internally which allows the device to be used with minimum external components.

The device has built-in protection features, such as cycle-by-cycle current limit, hiccup mode short-circuit protection, and thermal shutdown in case of excessive power dissipation. The TPS560430 is available in SOT-23-6 package.

### 2.3.10 TLV713 — 150-mA Low-Dropout (LDO) Regulator With Foldback Current Limit for Portable Devices

The TLV713 series of low-dropout (LDO) linear regulators are low quiescent current LDOs with excellent line and load transient performance and are designed for power-sensitive applications. These devices provide a typical accuracy of 1%.

The TLV713 series of devices is designed to be stable without an output capacitor. The removal of the output capacitor allows for a very small solution size. However, the TLV713 series is also stable with any output capacitor if an output capacitor is used.

The TLV713 also provides inrush current control during device power up and enabling. The TLV713 limits the input current to the defined current limit to avoid large currents from flowing from the input power source. This functionality is especially important in battery-operated devices.

The TLV713 series is available in standard DQN and DBV packages. The TLV713P provides an active pulldown circuit to quickly discharge output loads.

### 2.3.11 TMP61 — Small Silicon-Based Linear Thermistor for Temperature Sensing

The TMP61 series of small silicon linear thermistors are designed for temperature measurement, protection, compensation, and control systems. Compared to traditional NTC thermistors, the TMP61 device offers enhanced linearity and consistent sensitivity across the full temperature range. The TMP61 offers robust performance due to device immunity to environmental variation and built-in fail-safe behaviors at high temperatures. This device is currently available in a 2-pin, surface-mount, 0402 footprint-compatible X1SON package.

### 2.3.12 CSD18510Q5B — 40-V, N-Channel NexFET™ MOSFET, Single SON5x6, 0.96 mOhm

This 40-V, 0.79-mΩ, SON 5-mm × 6-mm NexFET™ power MOSFET has been designed to minimize losses in power conversion applications.

### 2.3.13 UCC28911 — 700-V Flyback Switcher With Constant-Voltage, Constant-Current, and Primary-Side Regulation

The UCC28910 and UCC28911 are high-voltage flyback switchers that provide output voltage and current regulation without the use of an optical coupler. Both devices incorporate a 700-V power FET and a controller that process operating information from the flyback auxiliary winding and power FET to provide a precise output voltage and current control. The integrated high-voltage current source for startup that is switched off during device operation, and the controller current consumption is dynamically adjusted with load. Both enable the very low stand-by power consumption.

Control algorithms in the UCC28910 and UCC28911, combining switching frequency and peak primary current modulation, allow operating efficiencies to meet or exceed applicable standards. Discontinuous conduction mode (DCM) with valley switching is used to reduce switching losses. Built-in protection features help to keep secondary and primary component stress levels in check across the operating range. The frequency jitter helps to reduce EMI filter cost.

### 2.3.14 SN74LVC1G3157DRYR — Single-Pole Double-Throw Analog Switch

This single channel single-pole double-throw (SPDT) analog switch is designed for 1.65-V to 5.5-V VCC operation. The SN74LVC1G3157 device can handle both analog and digital signals. The SN74LVC1G3157 device permits signals with amplitudes of up to VCC (peak) to be transmitted in either direction. Applications include signal gating, chopping, modulation or demodulation (modem), and signal multiplexing for analog-to-digital and digital-to-analog conversion systems.

## 2.4 System Design Theory

This reference design provides a universal AC mains-powered, 1000-W nominal output at 12 V and 84 A. This design comprises a front-end totem pole PFC power stage followed by an isolated DC/DC LLC power stage.

### 2.4.1 Totem Pole PFC Stage Design

The totem pole bridgeless PFC is an old structure that has been presented for many years, but it has not been applied in products because of the poor reverse recovery performance of the MOSFET's body diode. However, in recent years, benefitting from GaN HEMT's no reverse recovery feature, the totem pole PFC is now being applied and and has become a simple structure. Generally, the totem pole PFC has one fast-switching leg and one slow-switching leg. The slow-switching leg can use both a rectifier diode or MOSFET. When using a MOSFET as a slow-switching leg, the totem pole PFC can achieve higher efficiency and bidirectional power conversion between the AC side and the DC side.

#### *2.4.1.1 Design Parameters of the PFC Stage*

**Table 2-1. Design Parameters of the PFC Stage**

| SYMBOL | PARAMETER | MINIMUM | NOMINAL | MAXIMUM | UNIT |
|---|---|---|---|---|---|
| $V_{IN}$ | AC input voltage | 100 | 230 | 264 | V AC |
| $f_{line}$ | Line frequency | 47 | | 63 | |
| $V_{out}$ | Output voltage | | 385 | | V DC |
| $P_{out\ (nom)}$ | Output power | | | 1050 | Watt |
| η | Targeted efficiency | | 98.8% | | |
| iTHD | Targeted input current THD | | < 5% at >30% Load | | |
| PF | Targeted power factor | | 0.99 | | |
| $F_{sw}$ | Switching frequency | | 100 | | kHz |

#### *2.4.1.2 Current Calculations*

The input fuse, bridge rectifier, and input capacitor are selected based on the input current calculations. First, determine the maximum average output current, $I_{OUT(max)}$ as per Equation 1:

$$I_{OUT(max)} = \frac{P_{OUT(max)}}{V_{dc\_link}} = \frac{1050\ W}{385\ V} = 2.73\ A \quad (1)$$

The maximum input RMS line current, $I_{IN\_RMS(max)}$, is calculated using the parameters from Table 2-1, and the efficiency and power factor initial assumptions are calculated as follows:

$$I_{INrms(max)} = \frac{P_{OUT(max)}}{\eta_{PFC} \times V_{IN(min)} \times PF} = \frac{1050\ W}{98\% \times 100\ V \times 0.99} = 10.8\ A \quad (2)$$

#### *2.4.1.3 PFC Boost Inductor*

To determine the boost inductor, the maximum-allowed ripple current is calculated first. The maximum ripple current is observed at the lowest input voltage and maximum load. Assuming a maximum 40% ripple in the inductor current, the ripple current would be calculated as follows:

$$I_{INripple(max)} = \Delta ripple \times I_{INrms(max)} = 0.2 \times \sqrt{2} \times 10.7A = 6.06\ A \quad (3)$$

The duty cycle, $DUTY_{(max)}$, at the peak of the minimum input voltage can be calculated as:

$$DUTY_{(max)} = \frac{V_{OUT(max)} - V_{INmin \times \sqrt{2}}}{V_{dc\_link}} = \frac{385\ V - 100\ V \times \sqrt{2}}{385\ V} = 0.63 \quad (4)$$

The minimum value of the boost inductor is calculated based on the acceptable ripple current, at a worst-case duty cycle of 0.63:

$$L_{Boost} \geq V_{OUT(max)} \times DUTY_{(max)} \times \frac{1 - DUTY_{(max)}}{FSW \times I_{INripple(max)}} = 295\,\mu H \tag{5}$$

The actual value of the inductor chosen is 300 µH. The required saturation current for the boost inductor is calculated using Equation 6 for the minimum input voltage and short time maximum overload conditions.

$$I_{L(max)} = \sqrt{2} \times \frac{P_{OUT(max)}}{V_{INrms(min)}} \times (1 + \frac{\Delta ripple0}{2}) = 16.67\ A \tag{6}$$

#### 2.4.1.4 Output Capacitor

The output capacitor, $C_{bus}$, is sized to meet the DC link ripple and holdup requirements of the converter.

The ripple of DC link voltage can be calculated by Equation 7:

$$V_{ripple} = \frac{P_{OUT\times} 0.0032}{C_{bus} \times V_{Bus}} \tag{7}$$

The holdup time required by this design, $T_{Holdup}$, is 10 ms. The average bus voltage is set at 385 V, which is correlated with the LLC transformer ratio. Considering the DC link voltage has +/-10-V ripple at full load, the DC BUS voltage will be $V_{BUS(norm)}$ = 365 V in the valley point.

Considering the gain of LLC can not be too wide, the minimum input of LLC is set at $V_{BUS(min)}$ = 320 V. For energy, the minimum value of BUS capacitance can be calculated using Equation 8, and for the calculation of Equation 9, the available capacitance value of 680 uF was chosen.

$$C_{OUT} \geq 2 \times P_{OUT(nominal)} \times \frac{T_{Holdup}}{V^2_{OUT(norm)} - V^2_{OUT(min)}} \tag{8}$$

$$C_{OUT} \geq 2 \times 1000\ W \times \frac{10\ ms}{365\ V^2 - 320\ V^2} = 648\,\mu F \tag{9}$$

#### 2.4.1.5 Fast and Slow Switches

Given the calculation of $I_{IN\_RMS(max)}$, all current ratings of the switches should be greater than 10.8 A. In a CCM totem pole PFC, a GaN switch is required for the fast bridge. The 70-mΩ GaN LMG3410R070 was selected, which is recommended for 12-A $I_{DS}$ on 125°C $T_j$.

The slow bridge is only line frequency switching, so the minimum $R_{DS(on)}$ is preferred for better efficiency. To save space, the Toshiba® 50-mΩ MOSFET TK39A60W was chosen. This device has a low $R_{DS(on)}$ with a TO220 package that is available on the market.

#### 2.4.1.6 AC Current Sensing Circuits

In this design, AC current is sensed for current loop control. Only average current is required in the CCM boost PFC control loop. The TMCS1100, a 120-kHz Hall sensor, is a good option for this design. The TMCS1100 has high accuracy across a full temperature range and a very low offset current drift, helping achieve accuracy in input power metering.

The TMCS1100 has four options: 50 mV/A, 100 mV/A, 200 mV/A, and 400 mV/A. The supply voltage could be 3 V to 5.5 V. The offset voltage is set externally. 1.65 V was chosen for this design, which is half of the $V_{cc}$ of the controller. The maximum input current will reach 14 A at the peak value. In considering the worst case and leaving some margin, the TMCS1100A1 was chosen. The sensing circuits used the OPA376 with one analog switch, SN74LVC1G3157, to scale the voltage to match the ADC range at both the low line and high line. The circuits are shown in Figure 2-2.

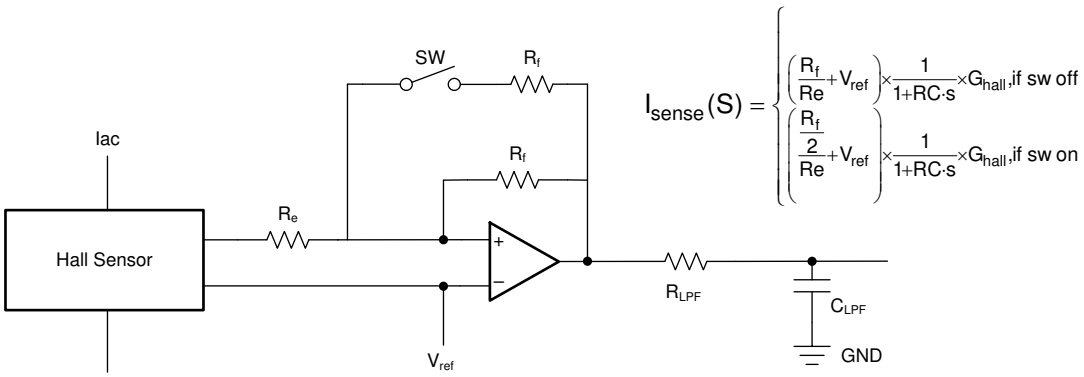



**Figure 2-2. Current Sensing and Signal Scaling Circuits**

#### *2.4.1.7 Temperature Sensing*

TI's silicon-based linear thermistor TPM61 is used for temperature sensing. The implementation circuits are shown in Figure 2-3.

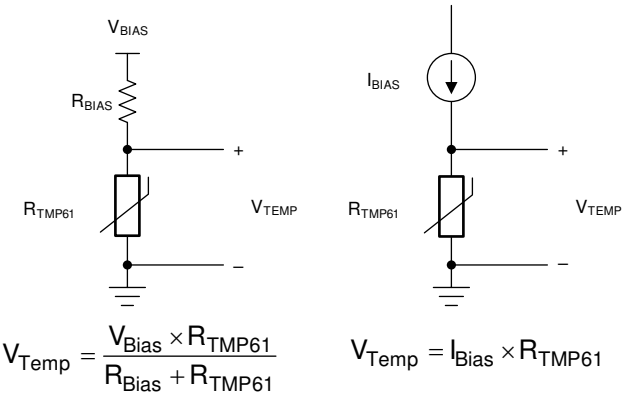



**Figure 2-3. Temperature Sensing Circuits**

### 2.4.2 Design Parameters of the LLC Stage

The LLC stage is the end stage for output regulation. Its input is the output of the PFC stage, so the LLC stage should be able to keep the output voltage stable at 12 V between the $V_{BUS(max)}$, the peak value with full load, and the $V_{BUS(min)}$, the minimum $V_{Bus}$ voltage for holdup time when AC drop occurs.

**Table 2-2. Design Parameters for the LLC Stage**

| SYMBOL | PARAMETER | MINIMUM | NOMINAL | MAXIMUM | UNIT |
|---|---|---|---|---|---|
| $V_{IN}$ | DC input voltage | 320 | 385 | 395 | V DC |
| $V_{out}$ | Output voltage | 11.4 | 12 | 12.6 | V DC |
| $P_{out}$ | Maximum output power | | | 1000 | Watt |
| $f_{sw}$ | Switch frequency | 80 | 140 | 250 | kHz |
| η | Targeted efficiency | | 98 | | % |

#### *2.4.2.1 Determining LLC Transformer Turns Ratio N*

The LLC tank is designed to have a nominal gain, Mg, of 1 at the resonant frequency. Use Equation 10 to estimate the required turns ratio.

$$n = \frac{V_{BUS(nom)}}{2 \times V_{out(norm)}} = \frac{385\ V}{2 \times 12.0\ V} = 16 \quad (10)$$

The transformer turns ratio is set to 16.

#### 2.4.2.2 Determining $M_{g\_min}$ and $M_{g\_max}$

Determine $M_{g\_min}$ and $M_{g\_max}$ using Equation 11 and Equation 12, respectively.

$$M_{g\_min} = n \times \frac{V_O + V_F}{V_{bus(max)} / 2} = 16 \times \frac{12\,V + 0.01\,V}{395\,V / 2} = 0.972 \quad (11)$$

$$M_{g\_max} = n \times \frac{V_O + V_F}{V_{bus(min)} / 2} = 16 \times \frac{12\,V + 0.01\,V}{320\,V / 2} = 1.2 \quad (12)$$

The dimensioned $M_{g\_max}$ is increased to one. One times the required value to have some margin is $M_{g\_max}$ = 1.1 × 1.2 ≈ 1.3.

#### 2.4.2.3 Determining Equivalent Load Resistance ($R_e$) of Resonant Network

Equation 13 calculates the equivalent load resistance at nominal and peak load under nominal output voltage and peak output voltage.

$$R_e = \frac{8 \times n^2}{\pi^2} \times \frac{V_O}{I_O} = \frac{8 \times 16^2}{\pi^2} \times \frac{12}{84} = 29.64\,\Omega \quad (13)$$

#### 2.4.2.4 Selecting $L_m$ and $L_r$ Ratio ($L_n$) and $Q_e$

Set the resonance point for the LLC converter close to 140 kHz to minimize the dimension of the LLC transformer set. The operating point of the LLC power stage is close to this frequency during a full load condition. Choose a value of $L_r$ = 19 µH and $C_r$ = 66 nF to calculate the value of the resonant frequency as follows:

$$f_r = \frac{1}{2\pi\sqrt{L_r C_r}} = 140\,kHz \quad (14)$$

A PQ 32/30 core has been used to realize the LLC transformer. The resonant inductance is provided through an additional shim inductor built on a 20/16 PQ core.

#### 2.4.2.5 Determining Primary-Side Currents

Use Equation 15 to calculate the primary-side RMS load current ($I_{pri}$) at a full load condition:

$$I_{pri} = \frac{\pi}{2\sqrt{2}} \times \frac{I_{o(nom)}}{n} = 1.11 \times \frac{84}{16} = 5.83\,A \quad (15)$$

As calculated in Equation 16, the RMS magnetizing current ($I_m$) at $f_{SW\_min}$ = 140 kHz is:

$$I_m = \frac{2\sqrt{2}}{\pi}\left(n \times \frac{V_{O(norm)} + V_F}{2\pi \times f_{sw(min)} \times L_m}\right) = 0.66\,A \quad (16)$$

Equation 17 calculates the resonant circuit current ($I_r$):

$$I_r = \sqrt{I^2_{pri} + I^2_m} = 5.86\,A \quad (17)$$

This value is also equal to the transformer primary winding current at $f_{SW\_min}$.

#### 2.4.2.6 Determining Secondary-Side Currents

The secondary-side RMS currents can be calculated from the average load current. Assuming that the LLC power stage is operating close to its second resonant frequency, the RMS current through each rectifier in the secondary-side push-pull output is calculated in Equation 18:

$$I_{sec\_RMS} = I_{sec} \times \frac{\pi}{2\sqrt{2}} = 92.56\ A \quad (18)$$

#### 2.4.2.7 Primary-Side GaN and Driver

Traditional LLC is basically achieved with Si MOSFETs. This reference design considers using GaN as a power tube because GaN is well-cooled, the input pin is voltage-compatible isolated drive, and can distinguish between overtemperature (OT) and overcurrent (OC). GaN also have advantages such as temperature reporting, 1-kW is sufficient with a 50-mR $R_{DS(on)}$ GaN.

To drive the primary-side GaN on the secondary side, use two ISO7721 isolators which can obtain the GaN fault information.

#### 2.4.2.8 Secondary-Side Synchronous MOSFETs

The output diode rectifier / SR MOSFET voltage rating is determined using Equation 19:

$$V_{dsmax} = 1.2 \times 2 \times V_{o(nom)} = 28.8\ V \quad (19)$$

The RMS currents through the output SR MOSFET is given with Equation 20:

$$I_{SRMOS\_RMS} = I_{sec\_RMS} \times \frac{1}{2} = \frac{92.56\ A}{2} = 46.28\ A \quad (20)$$

To reduce the losses in the synchronous rectifier, the 4-pcs CSD18510Q5B with very low $R_{DS(on)}$ and $Q_g$ is used in parallel. Each MOSFET used one dual-channel, non-isolated driver, the UCC27524, to reduce the conduction losses in the LLC output stage.

#### 2.4.2.9 Output Current Sensing

In this design, output current is sensed to optimize the system efficiency, adjust the deadtime and SR pulse, and is used for current sharing control between different PSUs. Also, this signal can be used in the control loop of the LLC stage to reduce the overshoot and undershoot during load transient.

A benefit of the high common-mode range of the INAx180 is the ability to put the sensing resistor either on high-side VO+ or low-side VO-. The INAx180 integrates a matched resistor gain network in four fixed-gain device options: 20, 50, 100, and 200. This matched gain resistor network minimizes gain error and reduces the temperature drift. To reduce the power loss on the current sensing resistor, the 200-V/V A4 devices should be chosen, and by adjusting the resistor to scale, the voltage to match the ADC range of the MCU. The sensing resistor can be calculated using Equation 21:

$$R_{CS} = \frac{V_{ADC}}{1.5 \times I_{omax} \times 200} = \frac{3.3\ V}{1.5 \times 84\ A \times 200} = 0.131\ m\Omega \quad (21)$$

When the resistor is smaller than 1 mΩ, the accuracy will drop and the cost will increase. Therefore, a 4-pcs, 0.5-mΩ resistor with 1% accuracy is used while getting 0.125 mΩ in parallel.

### 2.4.3 Communication Between the Primary Side and the Secondary Side

In this design, both the primary and secondary side use the F280049, F280039, or F280025. The Fast Serial Interface (FSI) communication available in the C2000 MCU is a good option for this design.

FSI originated as a solution for higher-bandwidth digital communication across the air gap, or hot-side to cold-side and vice versa, in high-voltage systems such as those used in industrial drives and digital power applications. FSI achieves the top clock rate of 50 MHz for $L_{VCMOS}$ IO and can take as few as two pins, CLK and Data, in each direction. FSI also has a dual-data rate. It latches the data on both the rising and falling clock edges, making the raw transmit bandwidth 100 Mbps and the raw receive bandwidth 100 Mbps as well.

The configuration of two signals in each direction that also requires reinforced isolation is addressed perfectly by another TI component, the ISO7742. A single SOIC16 packaged ISO7742 is all that is needed to isolate the 100 Mbps FSI signals up to 8000 Vpk and carries reinforced isolation certifications according to VDE, CSA, CQC, and TUV. This single chip, when using FSI, can replace the cost of multiple isolation devices while saving

significant board space and also reducing the PCB routing and voltage plane definition challenges associated with mixed-plane, high-voltage PCB designs.

# 3 Hardware, Software, Testing Requirements, and Test Results

## 3.1 Required Hardware and Software

### 3.1.1 Hardware

This section details the necessary equipment, test setup, and procedure instructions for the TIDA-010062 board testing and validation.

#### *3.1.1.1 Test Conditions*

For input, the power supply source ($V_{IN}$) must range from 100-V to 265-V AC. Set the input current limit of the input AC source to 15 A. For output, use an electronic variable load or a variable resistive load, which must be rated for ≥ 15 V and must vary the load current from 0 A to 90 A.

This section details the hardware and the different sections on the board. If only using the firmware of the design through powerSUITE, this section might not be valid. The key resources used for controlling the PFC stage on the MCU are listed in Table 3-1. The key resources used for controlling the LLC stage on the MCU are listed in Table 3-2. Figure 3-1 shows the key power stage and connectors on the design board.

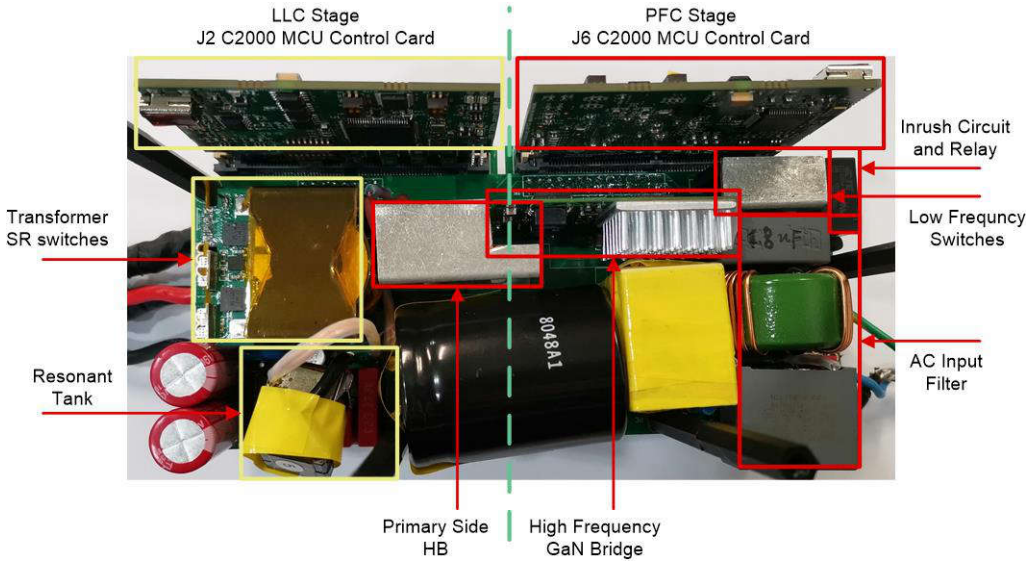



**Figure 3-1. Board Overview**

**Table 3-1. PFC Stage: Key Controller Peripherals Used for Control of Power Stage on Board**

| SIGNAL NAME | HSEC PIN NUMBER | FUNCTION |
|---|---|---|
| PWM-1A | 49 | PWM: low-frequency MOSFET leg, high-side switch |
| PWM-1B | 51 | PWM: low-frequency MOSFET leg, low-side switch |
| PWM-3A | 50 | PWM: high-frequency GaN leg, high-side switch. |
| PWM-3B | 52 | PWM: high-frequency GaN leg, low-side switch. |
| Iac | 18 | ADC with CMPSS: AC return current measurement |
| VL | 20 | ADC: AC voltage line |
| VN | 17 | ADC: AC voltage neutral |
| Vbus | 24 | ADC: bus voltage |
| In Rush Relay | 67 | GPIO: used to control the inrush relay |
| GaN Fault | 56 | GPIO: GaN fault signal |
| AC Current Sense GainChange | 63 | GPIO: controls the gain stage |

**Table 3-2. LLC Stage: Key Controller Peripherals Used for Control of Power Stage on Board**

| SIGNAL NAME | HSEC PIN NUMBER | FUNCTION |
|---|---|---|
| PWM-1A | 49 | High-side drive signal for primary side HB |
| PWM-1B | 51 | Low-side drive signal for primary side HB |
| PWM-2A | 53 | Drive signal for SRA |
| PWM-2B | 55 | Drive signal for SRB |
| $V_{OUT}$ | 25 | ADC with CMPSS: Output voltage measurement |
| $I_{OUT}$ | 15 | ADC with CMPSS: Resonant current measurement |
| $I_{reso}$ | 21 | ADC with CMPSS: Output current measurement |

#### *3.1.1.2 Test Equipment Required for Board Validation*

The following test equipment is required for board validation:

- Isolated AC source
- Single-phase power analyzer
- Digital oscilloscope
- High voltage probe (> 600 V)
- Current probe
- Multimeters
- Electronic or resistive load
- 12V1A cooling FAN

#### *3.1.1.3 Test Procedure*

##### 3.1.1.3.1 System Test: Dual Stages

1. Connect the GaN daughter cards and C2000 Piccolo controlCard to the mother board as follows. The details of the connections to each pin of the connectors can be found in the schematic of the *1-kW, 80+ Titanium AC/DC Power Supply Reference Design*.
   - One GaN daughter card 'LMG3410-HB-EVM' at connectors Brd1 position
   - C2000 Piccolo controlCard 'TMDSCNCD280049C'/'TMDSCNCD280025C' at the connector J1 and J3
2. Connect the 3 Pins AC terminal of reference board to the AC power source.
3. Connect output terminals with wires to the electronic load, and maintain correct polarity.
4. Put a cooling FAN at the AC side to keep the board in wind;
5. Set and maintain a small load of approximately 10 A.
6. Current and voltage probes can be connected to observe the input current, input voltage, and output voltages.

##### 3.1.1.3.2 PFC Stage Test

1. Insert the control card in the J6 slot.
2. Connect a 12-V, 1-A DC power supply at 12Vp, PGND1 net.
3. Turn the 12-V power supply ON. The LED on the control card lights up and indicates the device is powered.
4. To connect JTAG, use a USB cable from the control card and connect it into a host computer. Build and load the project.
5. Different labs have different setups:

   For Lab 1, a DC power supply can be connected to the input J7. A resistive load of approximately 500 Ω and 400 W should be connected to the output at HVBUS, PGND net.

   For Lab 2, a DC power supply can be connected to the input J7. A resistive load of approximately 100 Ω and 400 W should be connected to the output at HVBUS, PGND net.

   For Lab 3, a single-phase AC power supply can be connected to the input J7. A resistive load of approximately 500 Ω and 400 W should be connected to the output at HVBUS, PGND net.

   For Lab 4, a single-phase AC power supply can be connected to the input J7. A constant current load of approximately 0.2 A and 1500 W should be connected to the output at HVBUS, PGND net.

6. Current and voltage probes can be connected to observe the input current, input voltage, and output voltages, as shown in Figure 3-1.

#### 3.1.1.3.3 LLC Stage Test

1. Insert the control card in the J2 slot.
2. Connect a 12-V, 1-A DC power supply at 12Vp, PGND1 net.
3. Connect a 6-V, 1-A DC power supply at 6Vcc, V0 net.
4. Turn the 12-V and 6-V power supply ON. The LED on the control card lights up and indicates the device is powered.
5. To connect JTAG, use a USB cable from the control card and connect it into a host computer. Build and load the project.
6. Different labs have different setups:

   For Lab 1, a DC power supply can be connected to the HVBUS, PGND net. A constant current load of approximately 10 A should be connected to the output at the 12Vout, V0-out net.

   For Lab 2, a DC power supply can be connected to the HVBUS, PGND net. A constant current load of approximately 10 A should be connected to the output at the 12Vout, V0-out net.
7. Current and voltage probes can be connected to observe the resonant current, switching node voltage, SR drive signal and output voltages, as Figure 3-1 shows.

### 3.1.2 PFC Stage Software

The software of this design is available inside C2000Ware Digital Power SDK and is supported inside the powerSUITE framework.

#### *3.1.2.1 Opening Project Inside CCS*

To start:

1. Install CCS from the *Code Composer Studio (CCS) Integrated Development Environment (IDE)* tools folder, CCSV10.1 or above is recommended.
2. Install C2000Ware DigitalPower SDK at the *C2000Ware Digital Power SDK* tools folder.
   - Note: powerSUITE is installed with the SDK in the default install.
3. Go to *View* → *Resource Explorer*. Below the TI Resource Explorer, go to *C2000Ware DigitalPower SDK*.

   To open the reference design software as it is (opens firmware as it was run on this design and hardware, requires the board to be exactly the same as this reference design).
4. Under *C2000Ware DigitalPower SDK*, select *Development Kits* → *CCM Totem Pole PFC TIDA-010062*, and click on *Run <Import> Project*.
5. These steps import the project, and the development kit or designs page show up. This page can be used to browse all the information on the design including this user guide, test reports, hardware design files, and so forth.
6. Click *Import <device_name> Project*.
7. This action imports the project into the workspace environment, and a main.syscfg page with a GUI similar to Figure 3-3 shows up.

### 3.1.2.2 Project Structure

Once the project is imported, the project explorer appears inside CCS as shown in Figure 3-2.

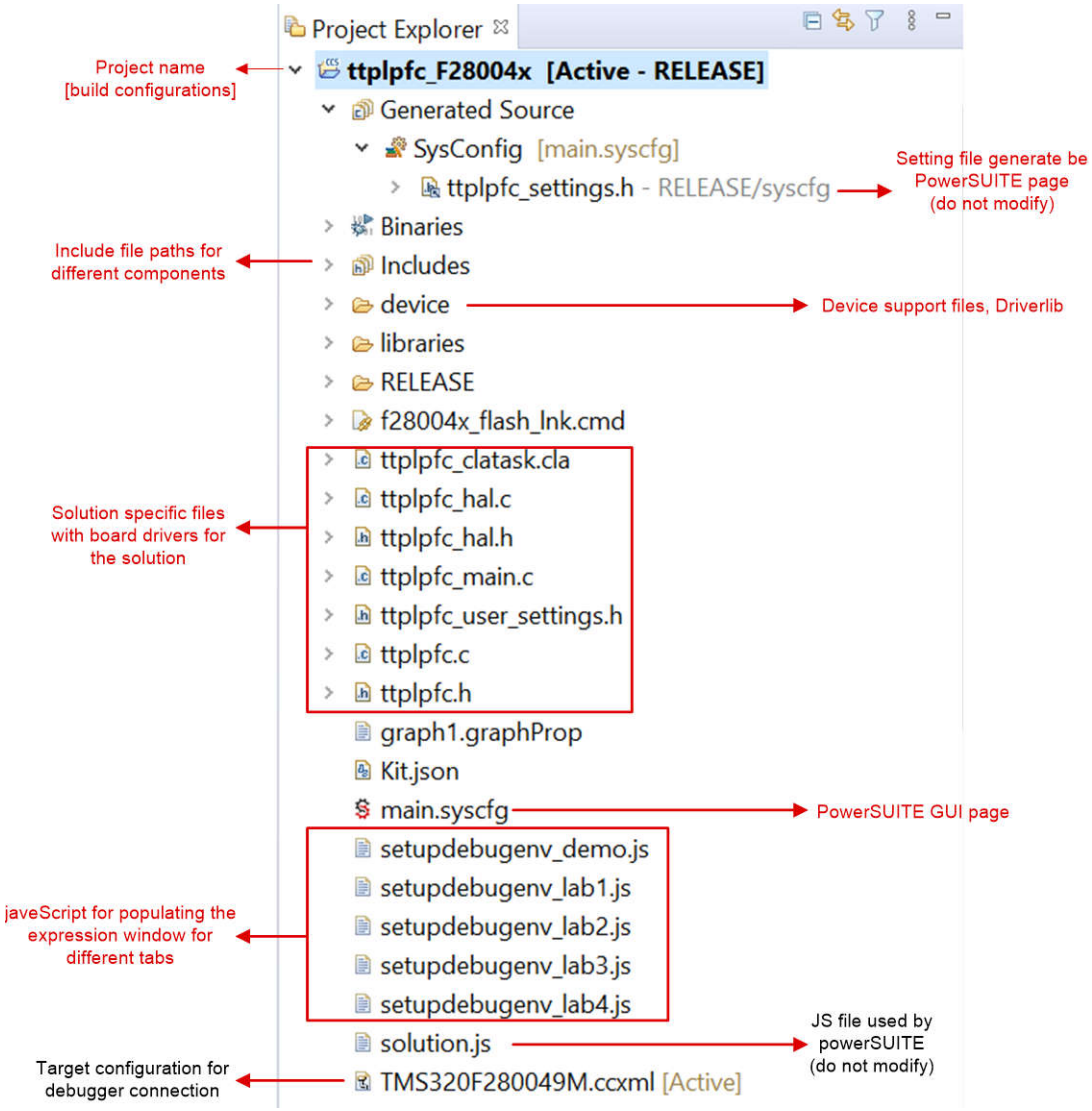



**Figure 3-2. Project Explorer View of PFC Solution Project**

Click main.syscfg file, a GUI page appears with modifiable options for the PFC solution (Figure 3-3). This GUI can be used to change the parameters for an adapted solution, like power rating, inductance, capacitance, sensing circuit parameters, and so forth.

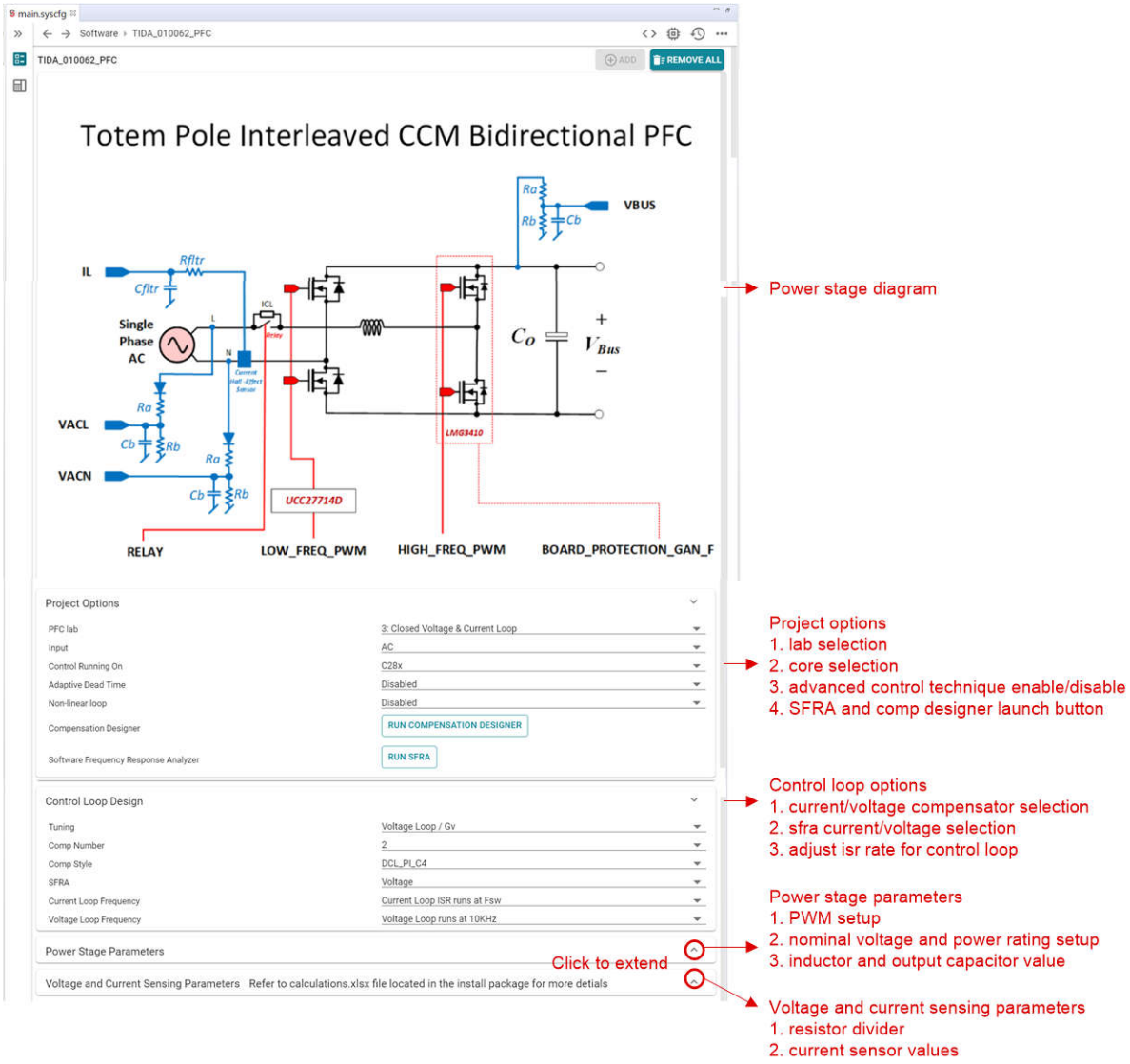



**Figure 3-3. powerSUITE Page for CCM TTPL PFC Stage**

The general structure of the project is shown in Figure 3-4.

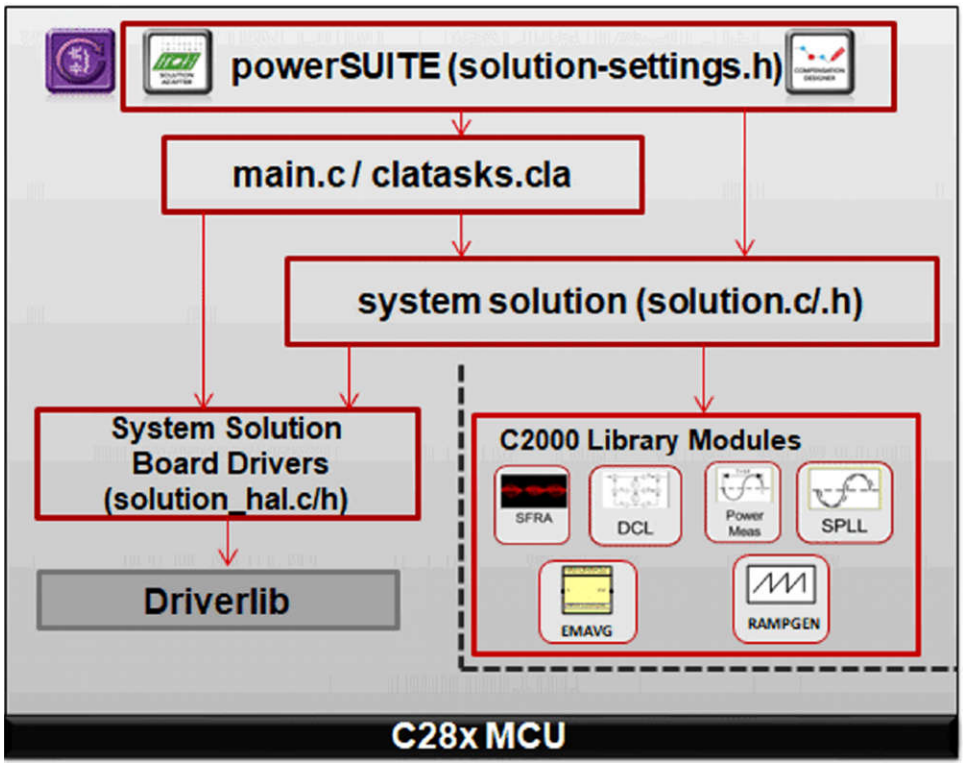



**Figure 3-4. Project Structure Overview**

Figure 3-4 shows the project for F28004x/F28002x/F28003x; however, if a different device is chosen from the powerSUITE page, the structure is similar.

Solution specific and device independent files are *<solution>.c/h*. This file consists of the main.c file of the project and is responsible for the control structure of the solution.

For this stage of this design <solution> is *ttplpfc*.

Board-specific and device-specific files are *<solution>_hal.c/h*. This file consists of device specific drivers to run the solution.

The powerSUITE page can be opened by clicking on the *main.syscfg* file, listed under the project explorer. The powerSUITE page generates the *<solution>_settings.h* file. This file is the only file used in the compile of the project that is generated by the powerSUITE page. The user must not modify this file manually as the changes are overwritten by powerSUITE every time the project is saved. User can modify several settings in *<solution>_user_settings.h* file.

The *Kit.json* and *solution.js* files are used internally by the powerSUITE and must also not be modified by the user. Any changes to these files result in project not functioning properly.

The setupdebugenv_build.js are provided to auto-populate the watch window variables for different labs.

The *.graphProp files is provided to auto populate settings for the data logger graph.

The project consists of an interrupt service routine, which is called every PWM cycle, and a current controller is run inside this ISR. In addition to this, there is a slower ISR of approximately 10 kHz that is called for running the voltage loop and the instrumentation ISR. A few background tasks (A0-A4 and B0-B4) are called in a polling fashion and can be used to run slow tasks for which absolute timing accuracy is not required, such as SFRA background and so on.

Figure 3-5 shows the software flow diagram of the firmware.

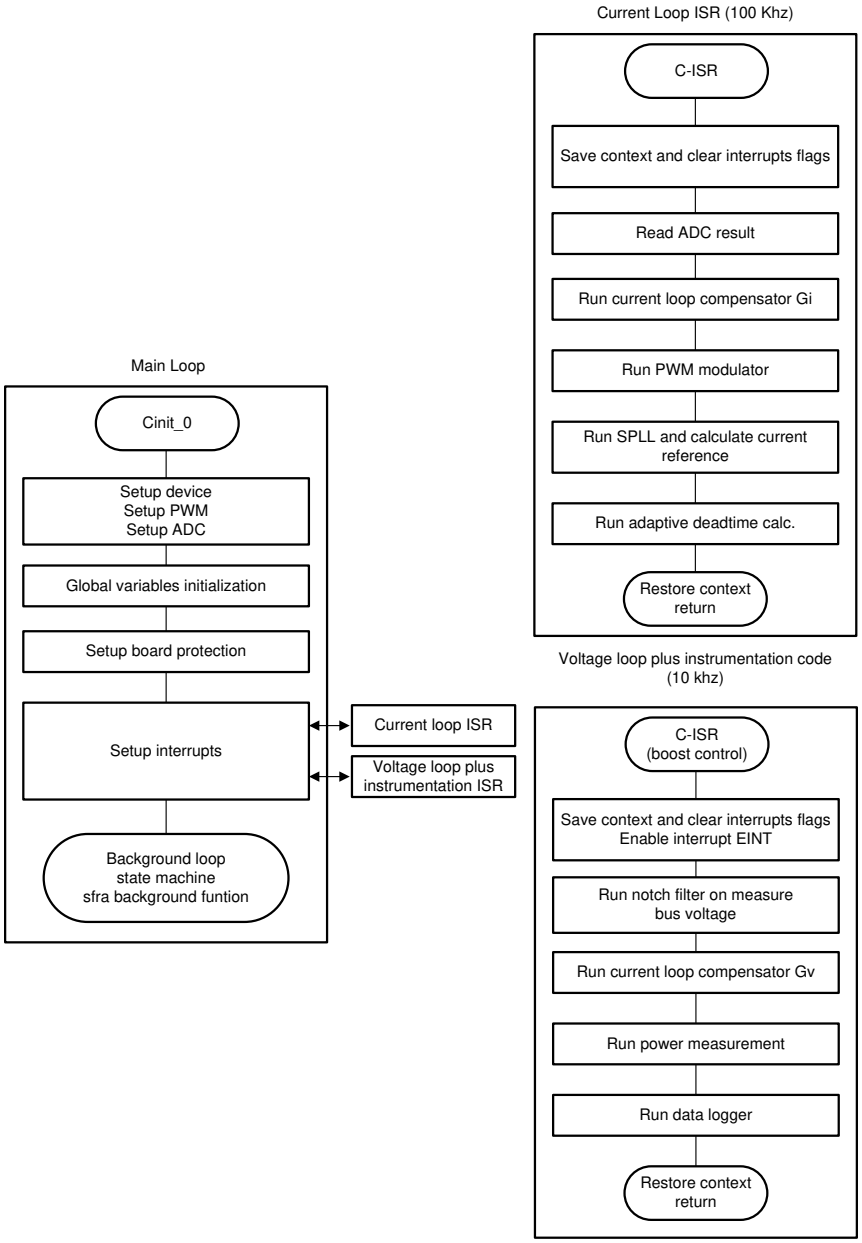



**Figure 3-5. PFC Stage Software Structure**

To simplify the system, bring up and design the software of this reference design is organized in 4 labs. The lab1 and lab2 are designed to validate Boost DC-DC operation. The lab3 and lab4 are designed to validate PFC AC-DC operation.

Lab 1: Open Loop, DC (PFC Mode)

Lab 2: Closed Current Loop DC

Lab 3: Closed Current Loop, AC (PFC)

Lab 4: Closed Voltage and Current Loop (PFC)

These labs are detailed in Section 3.1.2.5. If using the reference design hardware, make sure the hardware setup is completed as outlined in Section 3.1.1.

#### 3.1.2.3 Using CLA on C2000 MCU to Alleviate CPU Burden

The control law accelerator (CLA) is a co-processor available on the C2000 MCU family of devices. This coprocessor enables offloading the control-ISR functions from the main C28x CPU core.

To run the control ISR on the CLA for solutions supported in powerSUITE, selection is achieved through a drop-down menu on the powerSUITE CFG page. The software structure of the powerSUITE solution is designed such that offloading the task to the CLA is simply a drop-down menu selection. The code is not duplicated and a single source for the solution algorithm is maintained even when code is run on the CLA or the C28x. This configuration enables flexible debugging of the solution.

The CLA features of each device varies slightly. For example, on the F2837xD, F2837xS, and F2807x, the CLA can support only one task at a given time, and there is no nesting capability. This configuration means that the task cannot be interrupted. Only one ISR can be offloaded to the CLA. On the F28004x, the CLA supports a background task from which a regular CLA task can nest. This configuration enables offloading two ISRs on the CLA.

The CLA supports a background task from which it can nest into a CLA task. This configuration allows offloading two ISR functions to the CLA. For the F28003x/F28004x, both the control ISR (100 kHz) for the current loop and the voltage loop and instrumentation ISR (10 kHz) are offloaded to the CLA. For the F28002x which do not have CLA, both the ISRs can only run on C28x.

For more information on the CLA, visit the CLA Hands-On Workshop and the respective device technical reference manuals.

#### 3.1.2.4 CPU Utilization and Memory Allocation

The CPU utilization can be monitored by toggling GPIOs and capturing the waveforms using an oscilloscope. Each ISR includes profiling functions that set GPIO pin high at the beginning of ISR and set GPIO pin low at the end of ISR. ISR loadings with advanced options enabled can be measured in the same way by configuring main.syscfg.

| | ISR1 (100 kHz) | ISR2 (10 kHz) |
|---|---|---|
| CPU utilization | 42% | 10% |

Figure 3-6 shows the memory allocation (F28004x).

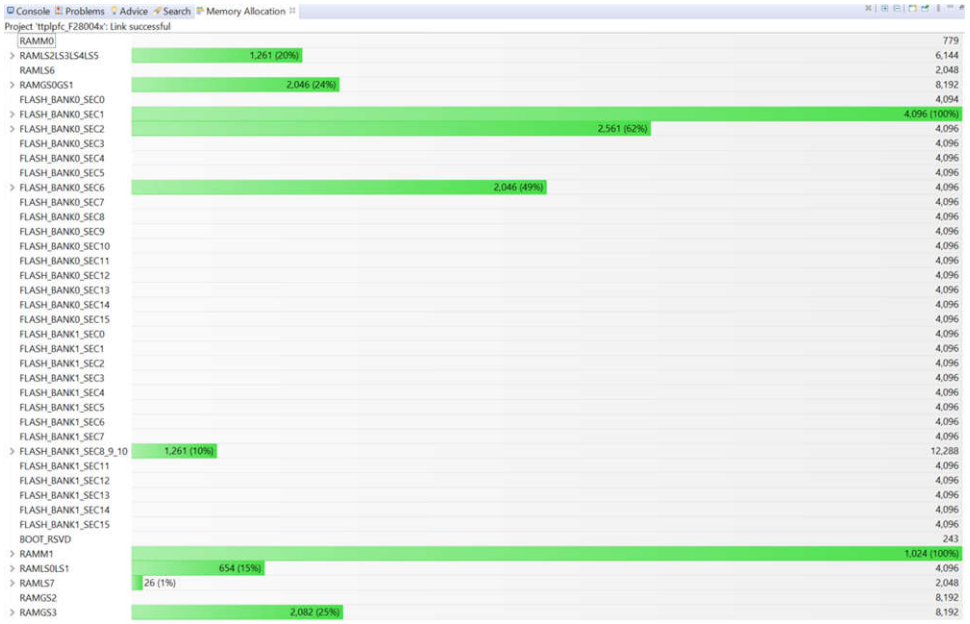



**Figure 3-6. TIDA-010062 PFC Stage Memory Allocation (F28004x)**

### 3.1.2.5 Running the Project

#### 3.1.2.5.1 Lab 1: Open Loop, DC (PFC Mode)

In this lab the board is excited in open loop fashion with a fixed duty cycle. To test lab 1, set the output load 500 Ω. The duty cycle is controlled with the dutyPU_DC variable. This lab verifies the sensing of feedback values from the power stage and also operation of the PWM gate driver and ensures there are no hardware issues. Additionally, calibration of input and output voltage sensing can be performed in this lab. Figure 3-7 shows the software structure for this lab. There are two ISRs in the system: fast ISR for the current loop and a slower ISR to run the voltage loop and instrumentation functions. Modules that are run in each ISR are shown in Figure 3-7.

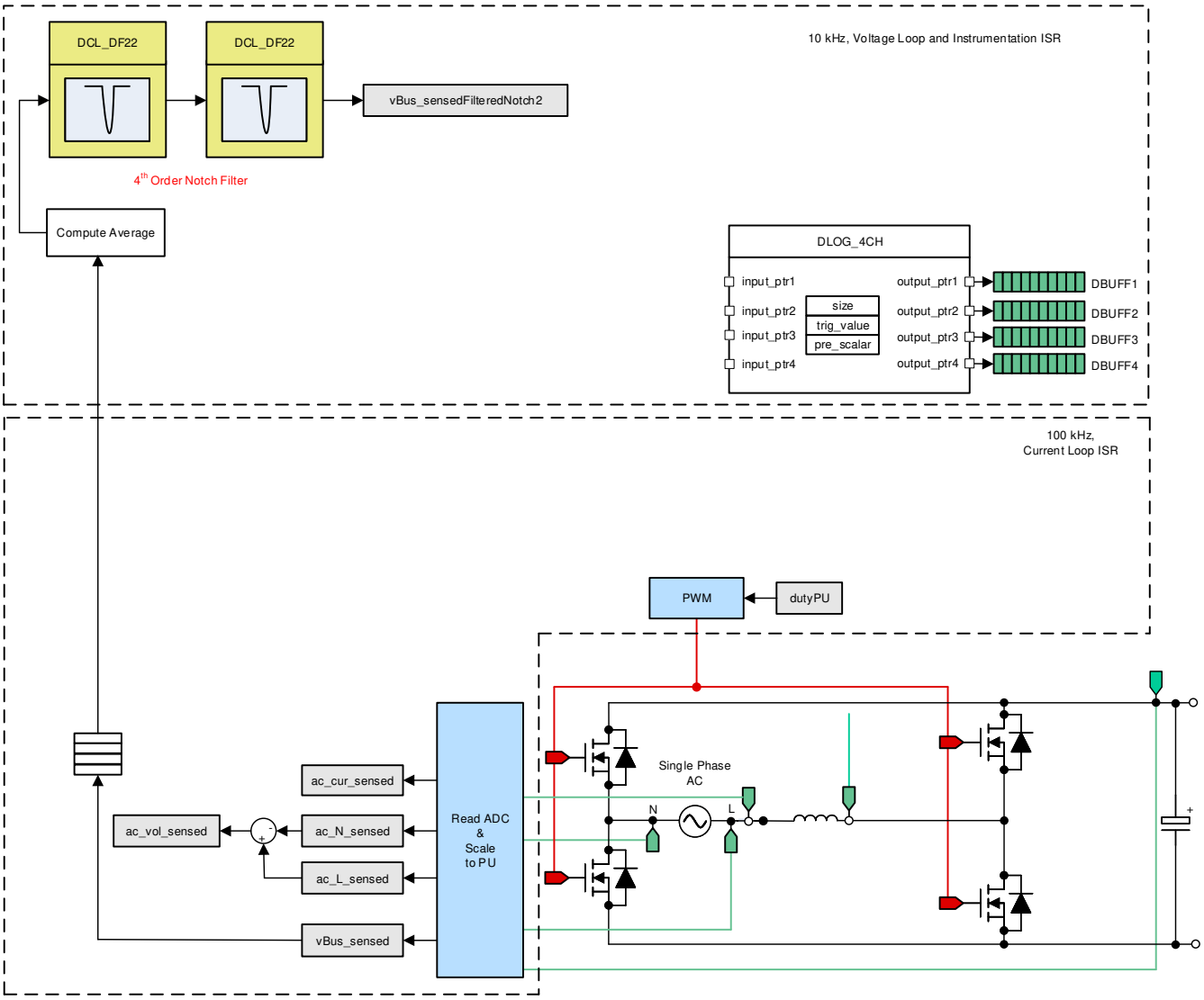



**Figure 3-7. PFC Lab 1 Control Software Diagram: Open Loop Project**

##### 3.1.2.5.1.1 Setting Software Options for Lab 1

1. powerSUITE Settings: On the powerSUITE page select under *Project Options* section:
   - Select *Lab1* under the Lab option.
   - Also disable the other options such as *Non-linear Voltage Loop, Adaptive Deadtime*
2. If this is an adapted solution, edit the setting under *Voltage and Current Sensing Parameters*
3. Under Power Stage Parameters specify the switching frequency, the dead band, and the power rating.
4. Save the page.

#### 3.1.2.5.1.2 Building and Loading Project

1. Right click on the project name, and click *Rebuild Project*.
2. The project builds successfully.
3. In the *Project Explorer* make sure the correct target configuration file is set as Active under targetconfigs (Figure 3-2).
4. Then click *Run* → *Debug* or debug button . This action launches a debugging session. In the case of dual CPU devices, a window may appear to select the CPU that the debug must be performed. In this case, select CPU1.
5. The project then loads on the device, and CCS debug view becomes active. The code halts at the start of the main routine.

#### 3.1.2.5.1.3 Setup Debug Environment Windows

1. To add the variables in the watch and expressions window, click *View* → *Scripting Console* to open the scripting console dialog box. On the upper-right corner of this console, click on open then browse to the *setupdebugenv_lab1.js* script file located inside the project folder. This script file populates the watch window with appropriate variables required to debug the system. Click on the *Continuous Refresh* button on the watch window to enable continuous update of values from the controller. The watch window appears as shown in Figure 3-8.

Variables | Expressions | Registers

| Expression | Type | Value | Address |
|---|---|---|---|
| TTPLPFC_lab.enum_ | enum <unnamed> | Lab1 | 0x00008080@Data |
| TTPLPFC_guiVbus_V | float | 3.37148213 | 0x00008004@Data |
| TTPLPFC_guiVin_Vol | float | 2.88032961 | 0x00008018@Data |
| TTPLPFC_guiIL_Amp | float | -0.163826719 | 0x0000806E@Data |
| TTPLPFC_ac_cur_sen | float | -0.00970900059 | 0x00008024@Data |
| TTPLPFC_iLMeasOff | float | 0.497999996 | 0x0000802C@Data |
| EPwm1Regs.TZFLG | Register | 0x0000 | |
| EPwm3Regs.TZFLG | Register | 0x0000 | |
| TTPLPFC_dutyPU | float | 0.5 | 0x00008048@Data |
| TTPLPFC_dutyPU_D | float | 0.5 | 0x0000804A@Data |
| TTPLPFC_vBus_sens | float | 0.00732421875 | 0x00008032@Data |
| TTPLPFC_autoStartS | long | 101 | 0x00008054@Data |
| Add new expression | | | |

**Figure 3-8. PFC Lab 1 Expressions View**

2. Run the project by clicking on .
3. Now Halt the processor by using the *Halt* button on the toolbar ( ).

#### 3.1.2.5.1.4 Using Real-Time Emulation

Real-time emulation is a special emulation feature that allows windows within CCS to be updated while the MCU is running. This feature allows graphs and watch views to update but also allows the user to change values in watch or memory windows and see the effect of these changes in the system without halting the processor.

1. Enable real-time mode by hovering the mouse on the buttons on the horizontal toolbar and clicking the button.
2. A message box may appear. If so, select *YES* to enable debug events. This action sets bit 1 (DGBM bit) of status register 1 (ST1) to a *0*. The DGBM is the debug enable mask bit. When the DGBM bit is set to *0*, memory and register values can be passed to the host processor for updating the debugger windows.

#### 3.1.2.5.1.5 Running Code

1. Run the project again by clicking on .
2. In a few seconds the inrush relay clicks, the software is programmed to do so in the lab1. The trip clears, and a duty cycle of 0.5 is applied.
3. In the watch view, check if the TTPLPFC_guiVin_Volts, TTPLPFC_guiVbus_Volts, TTPLPFC_guiiL_Amps, variables are updating, periodically.

**Note**

As no power is applied right now, this value is close to zero.

4. Slowly increase the input DC voltage from zero to 120 V. The output voltage shows a boosted voltage as a steady duty cycle of 0.5 PU is applied as default setting. If a high current is drawn, verify if the voltage terminals are swapped. If true, reduce the voltage to zero first and correct the issue before resuming the test.
5. Verifying the voltage sensing: Make sure *TTPLPFC_guiVin_Volts* and *TTPLPFC_guiVbus_Volts* display the correct values. This verifies the voltage sensing of the board in some manner.
6. Verifying the current sensing: Notice the *TTPLPFC_guiiL_Amps* for the given test condition.

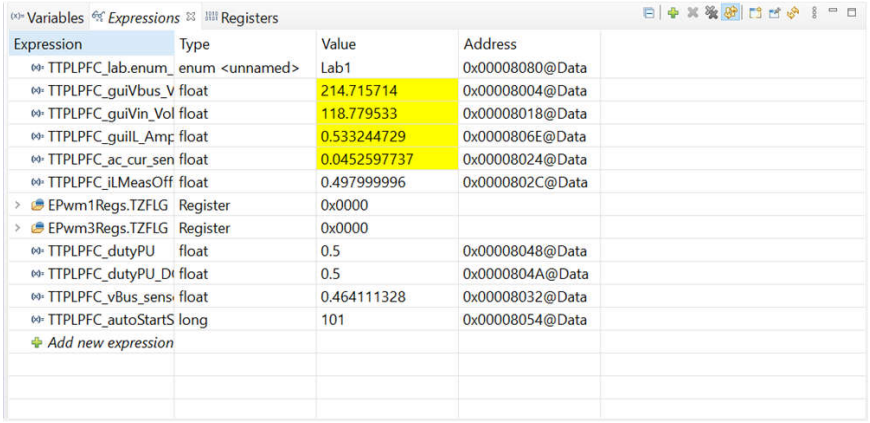



**Figure 3-9. PFC Lab 1: Watch Expression Showing Measured Voltage and Currents**

7. This verifies at a basic level the PWM driver and connection of hardware, the user can change the dutyPU_DC variable to see operation under various boost conditions.
8. Once finished, reduce the input voltage to zero and watch for the bus voltages to reduce down to zero.
9. This completes the check for this build, the following items are verified on successful completion of this build:
   - Sensing of voltages and currents and scaling to be correct
   - Interrupt generation and execution of the Lab 1 code in the current loop ISR and Voltage Loop Instrumentation ISR
   - PWM driver and switching

   If any issue is observed a careful inspection of the hardware may be required to eliminate any build issues, and so forth.
10. The controller can now be halted, and the debug connection terminated.
11. Fully halting the MCU when in real-time mode is a two-step process. First halt the processor by using the *Halt* button on the toolbar ( ) or by using *Target* → *Halt*. Then take the MCU out of real-time mode by clicking on . Finally, reset the MCU by clicking on .
12. Close CCS debug session by clicking on *Terminate Debug Session* (*Target* → *Terminate all*) .

#### 3.1.2.5.2 Lab 2: Closed Current Loop DC

In this build, Lab 2, the inner current loop is closed; that is, the inductor current is controlled using a current compensator Gi. To test lab 2, set the output load 100 Ω. Both DC bus and output voltage feed forward are applied to the output of this current compensator to generate the duty cycle of the inverter, Equation 22.

$$\text{duty1PU} = \frac{(\text{ac\_cur\_meas} - \text{ac\_cur\_ref\_inst}) \times \text{Gi} + \text{ac\_vol\_sensed}}{\text{vBus\_sensed}} \tag{22}$$

The complete software diagram for this lab is illustrated in Figure 3-10.

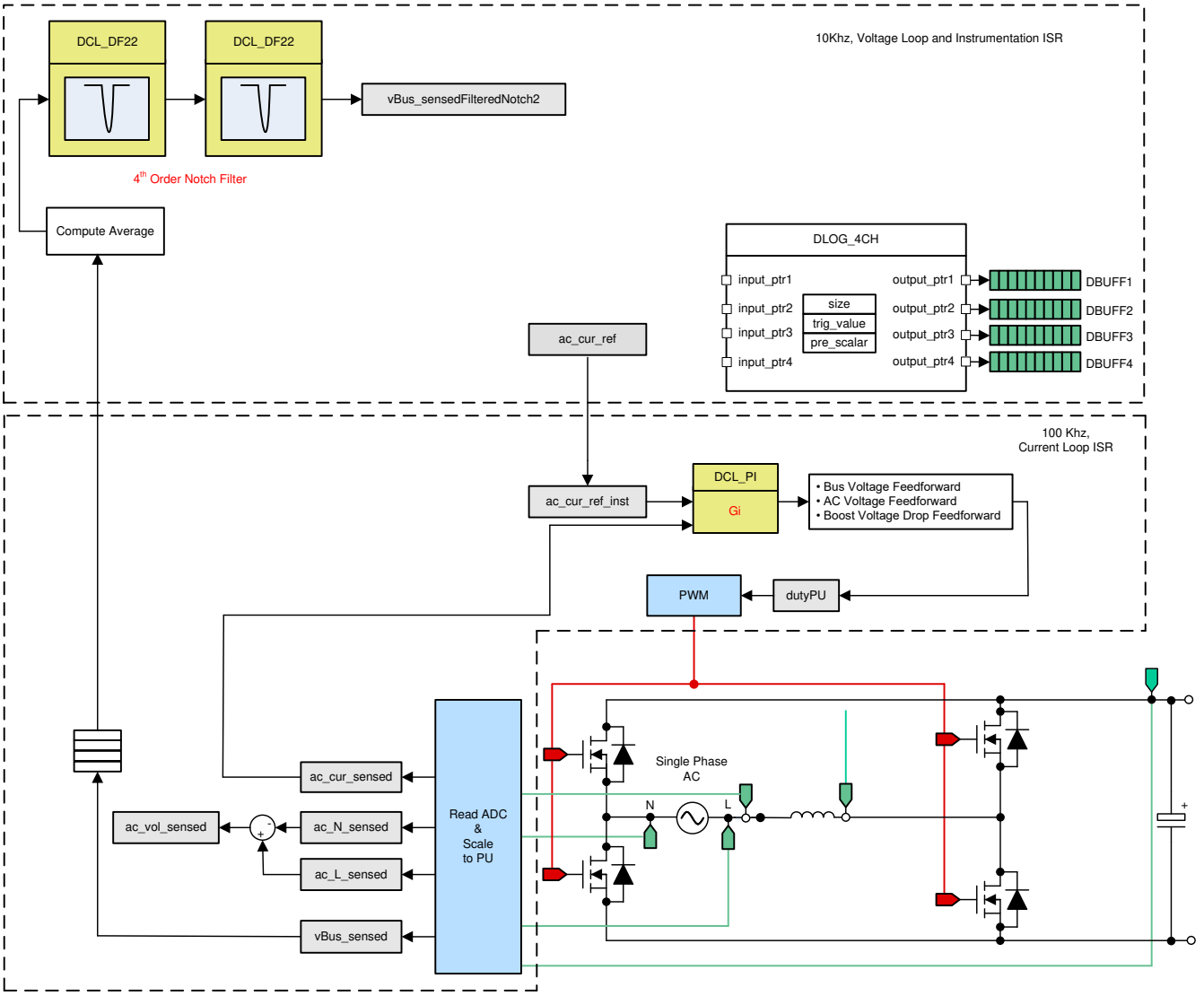



**Figure 3-10. PFC Lab 2 Control Software Diagram: Closed Current Loop**

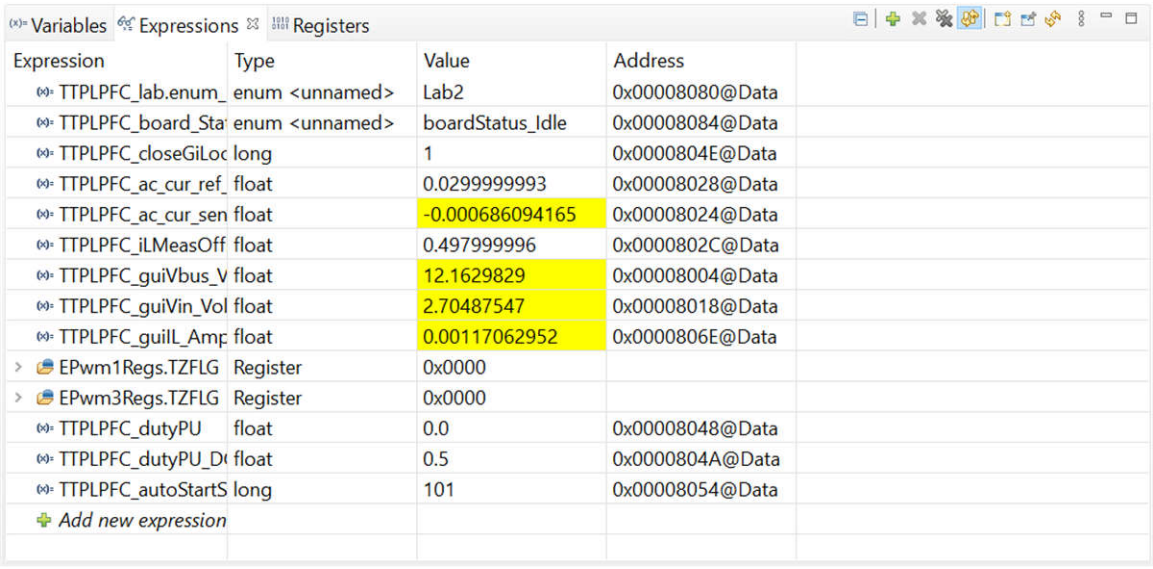
#### *3.1.2.5.2.1 Setting Software Options for Lab 2*

1. PowerSUITE settings: On the powerSUITE page select under *Project Options* section:
   - Select *lab2* under Lab option.
   - Select input to be DC under INPUT options
   - Also disable the other options such as *Non-linear Voltage Loop, Adaptive Deadtime*
2. Assume all other options are the same as specified in Section 3.1.2.5.1.

#### *3.1.2.5.2.2 Building and Loading Project and Setting up Debug*

1. Right click on the project name, and select *Rebuild Project*. The project builds successfully. Select *Run →Debug* or the debug button , which launches a debugging session. In the case of dual CPU devices, a window may appear to select the CPU the debug must be performed. In this case, select CPU1. The project then loads on the device, and CCS debug view becomes active. The code halts at the start of the main routine.
2. To add the variables in the watch and expressions window click *View → Scripting Console* to open the scripting console dialog box. On the upper right corner of this console, click on *Open* to browse to the *setupdebugenv_lab2.js* script file, which is located inside the project folder. This file populates the watch window with appropriate variables required to debug the system. Click on *Continuous Refresh* button ( )

on the watch window to enable continuous update of values from the controller. The watch window appears as shown in Figure 3-11.



Variables | Expressions | Registers

| Expression | Type | Value | Address |
|---|---|---|---|
| TTPLPFC_lab.enum_ | enum <unnamed> | Lab2 | 0x00008080@Data |
| TTPLPFC_board_Sta | enum <unnamed> | boardStatus_Idle | 0x00008084@Data |
| TTPLPFC_closeGiLoc | long | 1 | 0x0000804E@Data |
| TTPLPFC_ac_cur_ref_ | float | 0.0299999993 | 0x00008028@Data |
| TTPLPFC_ac_cur_sen | float | -0.000686094165 | 0x00008024@Data |
| TTPLPFC_iLMeasOff | float | 0.497999996 | 0x0000802C@Data |
| TTPLPFC_guiVbus_V | float | 12.1629829 | 0x00008004@Data |
| TTPLPFC_guiVin_Vol | float | 2.70487547 | 0x00008018@Data |
| TTPLPFC_guiIL_Amp | float | 0.00117062952 | 0x0000806E@Data |
| EPwm1Regs.TZFLG | Register | 0x0000 | |
| EPwm3Regs.TZFLG | Register | 0x0000 | |
| TTPLPFC_dutyPU | float | 0.0 | 0x00008048@Data |
| TTPLPFC_dutyPU_D | float | 0.5 | 0x0000804A@Data |
| TTPLPFC_autoStartS | long | 101 | 0x00008054@Data |
| *Add new expression* | | | |

**Figure 3-11. PFC Lab 2: Closed Current Loop Expressions View**

3. Enable real-time mode by hovering the mouse on the buttons on the horizontal toolbar, and clicking the button.
4. Run the project by clicking on .
5. Now Halt the processor by using the *Halt* button on the toolbar ( ).

#### *3.1.2.5.2.3 Running Code*

1. The project is programmed to drive the inrush relay and clear the trip after a set amount of time, that is, autoStartSlew==100. The software is programmed to do so in this lab. An input voltage must be applied after selecting run and before this autoslew counter reaches 100. If the counter reaches 100, before voltage is applied at the input, the code must be reset. For which the controller must be brought out of real time mode, a reset performed and restarted. Repeat steps from 3.
2. Now run the project by clicking .
3. Apply an input voltage of approximately 30 V before the TTPLPFC_autoStartSlew reaches 100. As soon TTPLPFC_autoStartSlew reaches 100, the inrush relay is triggered, and PWM trip is cleared along with closing the current loop flag.

Variables | Expressions | Registers

| Expression | Type | Value | Address |
|---|---|---|---|
| TTPLPFC_lab.enum_lab | enum <unnamed> | Lab2 | 0x00008080@Data |
| TTPLPFC_board_Status.enum_b | enum <unnamed> | boardStatus_Idle | 0x00008084@Data |
| TTPLPFC_closeGiLoop | long | 1 | 0x0000804E@Data |
| TTPLPFC_ac_cur_ref_pu | float | 0.0299999993 | 0x00008028@Data |
| TTPLPFC_ac_cur_sensed_pu | float | 0.0642700195 | 0x00008024@Data |
| TTPLPFC_iLMeasOffset_pu | float | 0.497999996 | 0x0000802C@Data |
| TTPLPFC_guiVbus_Volts | float | 46.1083298 | 0x00008004@Data |
| TTPLPFC_guiVin_Volts | float | 30.5685101 | 0x00008018@Data |
| TTPLPFC_guiIL_Amps | float | 0.783910036 | 0x0000806E@Data |
| EPwm1Regs.TZFLG | Register | 0x0000 | |
| EPwm3Regs.TZFLG | Register | 0x0000 | |
| TTPLPFC_dutyPU | float | 0.26251784 | 0x00008048@Data |
| TTPLPFC_dutyPU_DC | float | 0.5 | 0x0000804A@Data |
| TTPLPFC_autoStartSlew | long | 101 | 0x00008054@Data |
| *Add new expression* | | | |

**Figure 3-12. PFC Lab 2: Watch Expression, After Closed Current Loop Operation Begins**

4. The input current regulates around 0.7 A, and the output voltage boosts to approximately 46 V.
5. This action verifies the current compensator design.
6. To bring the system to a safe stop, bring the input DC voltage down to zero, observe the TTPLPFC_guiVBus_Volts comes down to zero as well.

   Fully halting the MCU when in real-time mode is a two-step process. First halt the processor by using the *Halt* button on the toolbar ( ) or by using *Target* → *Halt*. Then take the MCU out of real-time mode by clicking on . Finally, reset the MCU by clicking on .
7. Close CCS debug session by clicking on *Terminate Debug Session* (*Target*→*Terminate all*) .

#### 3.1.2.5.3 Lab 3: Closed Current Loop, AC (PFC)

In Lab 3 the inner current loop is closed; that is, the inductor current is controlled using a current compensator Gi. To test lab 3, set the output load 500 Ω. Both DC bus and output voltage feed-forward are applied to the output of this current compensator++ to generate the duty cycle of the inverter along with soft start for PWM around the zero crossing.

The complete software diagram for this lab is illustrated in Figure 3-13.

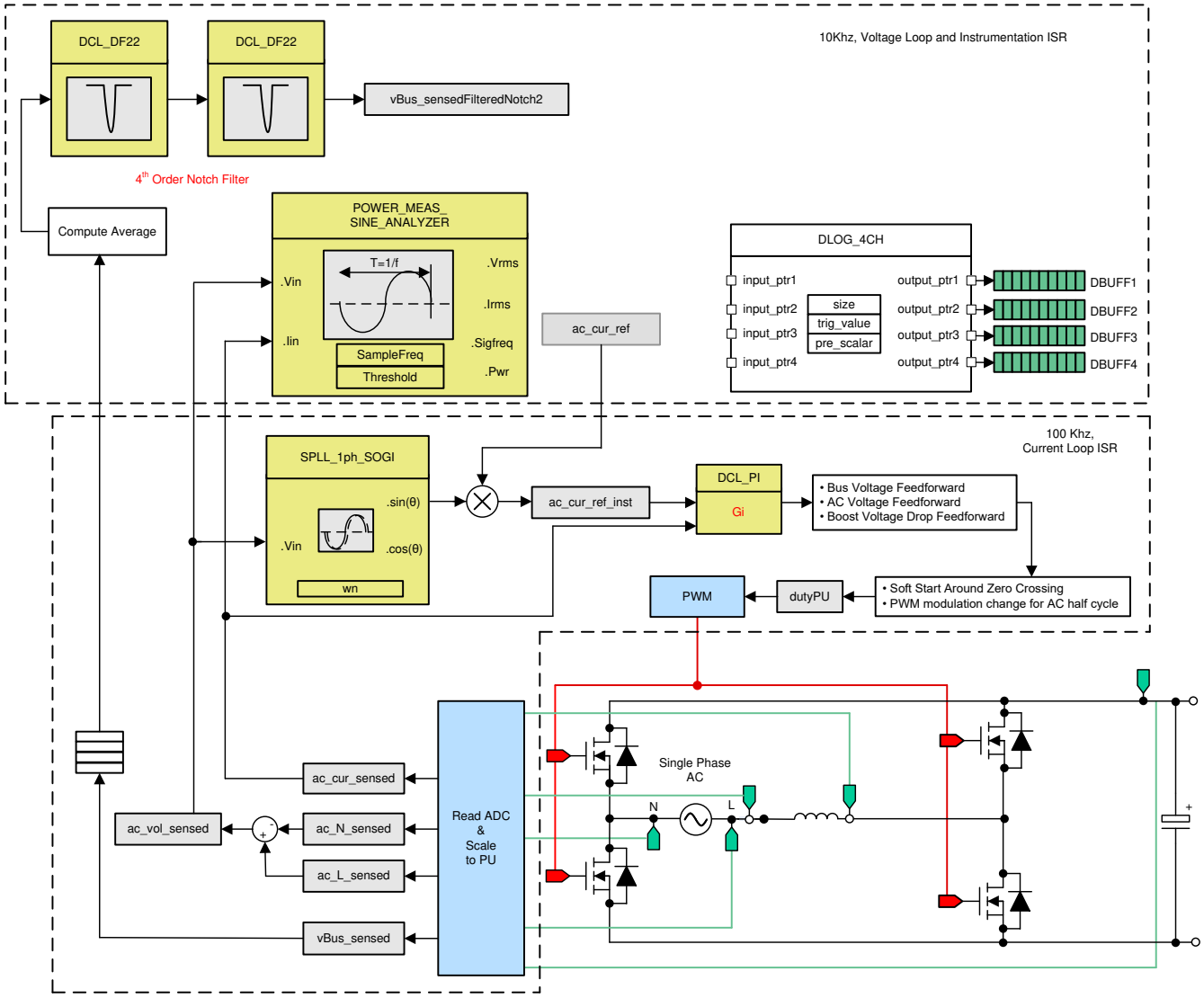



**Figure 3-13. PFC Lab 3 Control Software Diagram: Closed Current Loop AC**

##### *3.1.2.5.3.1 Setting Software Options for Lab 3*

1. PowerSUITE settings: On the powerSUITE page select under *Project Options* section:
   - Select *lab3* under Lab option
   - Select input to be AC under INPUT options
   - Also disable the other options such as *Non-linear Voltage Loop, Adaptive Deadtime*
2. Current compensator from the previous build is re-used in this lab so no additional steps are required for tuning the current loop in the lab.

##### *3.1.2.5.3.2 Building and Loading Project and Setting up Debug*

1. Right click on the project name, and click *Rebuild Project*. The project builds successfully. Click *Run →Debug* or the debug button , which launches a debugging session. In the case of dual CPU devices,

a window may appear to select the CPU the debug must be performed. In this case, select CPU1. The project then loads on the device, and CCS debug view becomes active. The code halts at the start of the main routine.

2. To add the variables in the watch and expressions window click *View* → *Scripting Console* to open the scripting console dialog box. On the upper right corner of this console, click on *Open* to browse to the *setupdebugenv_lab3.js* script file, which is located inside the project folder. This file populates the watch window with appropriate variables required to debug the system. Click on the *Continuous Refresh* button ( ) on the watch window to enable continuous update of values from the controller.

Variables | Expressions | Registers

| Expression | Type | Value | Address |
|---|---|---|---|
| TTPLPFC_lab.enum_lab | enum <unnamed> | Lab3 | 0x00008080@Data |
| TTPLPFC_pwm_SwState.enum_ | enum <unnamed> | pwmSwState_defau... | 0x00008086@Data |
| TTPLPFC_board_Status.enum_b | enum <unnamed> | boardStatus_Input... | 0x00008084@Data |
| TTPLPFC_closeGiLoop | long | 0 | 0x0000804E@Data |
| TTPLPFC_ac_cur_ref_pu | float | 0.0299999993 | 0x00008028@Data |
| TTPLPFC_ac_cur_sensed_pu | float | -0.00336401165 | 0x00008024@Data |
| TTPLPFC_iLMeasOffset_pu | float | 0.497999996 | 0x0000802C@Data |
| TTPLPFC_guiVbus_Volts | float | 2.4508903 | 0x00008004@Data |
| TTPLPFC_guiVin_Volts | float | 2.31311536 | 0x00008018@Data |
| TTPLPFC_guiVrms_Volts | float | 0.0 | 0x00008008@Data |
| TTPLPFC_guiIrms_Amps | float | 0.0 | 0x0000800E@Data |
| TTPLPFC_guiPrms_W | float | 0.0 | 0x0000800C@Data |
| TTPLPFC_guiPowerFactor | float | 0.0 | 0x00008070@Data |
| EPwm1Regs.TZFLG | Register | 0x0004 | |
| EPwm3Regs.TZFLG | Register | 0x0004 | |
| TTPLPFC_dutyPU | float | 0.00999999978 | 0x00008048@Data |
| TTPLPFC_dutyPU_DC | float | 0.5 | 0x0000804A@Data |
| TTPLPFC_autoStartSlew | long | 0 | 0x00008054@Data |
| *Add new expression* | | | |

**Figure 3-14. PFC Lab 3: Closed Current Loop Expressions View**

3. Enable real-time mode by hovering the mouse on the buttons on the horizontal toolbar, and clicking the *Continuous Refresh* button .

#### *3.1.2.5.3.3 Running Code*

1. The project is programmed to wait for input voltage to exceed approximately 75 $V_{RMS}$ to drive the inrush relay, and clear the trip.
2. Run the project by clicking the button.
3. Now apply an input voltage of approximately 120 V, the board comes out of the undervoltage condition and inrush relay is driven. The trip clears, and a small amount of current of approximately 0.55 A RMS is drawn. The bus voltage is close to 180 V.
4. Slowly increase TTPLPFC_ac_cur_ref_pu to 0.14, that is, 2.4-A input, and the bus voltage rises to 343 V. Figure 3-15 shows the voltage and current waveform.

Variables | Expressions | Registers

| Expression | Type | Value | Address |
|---|---|---|---|
| TTPLPFC_lab.enum_lab | enum <unnamed> | Lab3 | 0x00008080@Data |
| TTPLPFC_pwm_SwState.enum_ | enum <unnamed> | pwmSwState_positi... | 0x00008086@Data |
| TTPLPFC_board_Status.enum_b | enum <unnamed> | boardStatus_NoFault | 0x00008084@Data |
| TTPLPFC_closeGiLoop | long | 1 | 0x0000804E@Data |
| TTPLPFC_ac_cur_ref_pu | float | 0.140000001 | 0x00008028@Data |
| TTPLPFC_ac_cur_sensed_pu | float | -0.0872333944 | 0x00008024@Data |
| TTPLPFC_iLMeasOffset_pu | float | 0.497999996 | 0x0000802C@Data |
| TTPLPFC_guiVbus_Volts | float | 343.971344 | 0x00008004@Data |
| TTPLPFC_guiVin_Volts | float | -143.850204 | 0x00008018@Data |
| TTPLPFC_guiVrms_Volts | float | 119.446823 | 0x00008008@Data |
| TTPLPFC_guiIrms_Amps | float | 2.51879764 | 0x0000800E@Data |
| TTPLPFC_guiPrms_W | float | 301.779388 | 0x0000800C@Data |
| TTPLPFC_guiPowerFactor | float | 0.980018437 | 0x00008070@Data |
| EPwm1Regs.TZFLG | Register | 0x0000 | |
| EPwm3Regs.TZFLG | Register | 0x0000 | |
| TTPLPFC_dutyPU | float | 0.0559171699 | 0x00008048@Data |
| TTPLPFC_dutyPU_DC | float | 0.5 | 0x0000804A@Data |
| TTPLPFC_autoStartSlew | long | 5 | 0x00008054@Data |
| Add new expression | | | |

**Figure 3-15. Watch Expression, Lab 3, AC After Closed Current Loop Operation Begins**

5. SFRA is integrated in the software of this lab to verify the designed compensator provides enough gain and phase margin by measuring on hardware. To run the SFRA keep the project running, and from the powerSUITE page, click on the *SFRA* icon. The SFRA GUI appears.
6. Select the options for the device on the SFRA GUI. For example, for F280049M select floating point. Click on *Setup Connection*. On the pop-up window uncheck the boot on connect option, and select an appropriate COM port. Click *OK*. Return to the SFRA GUI, and click *Connect*.

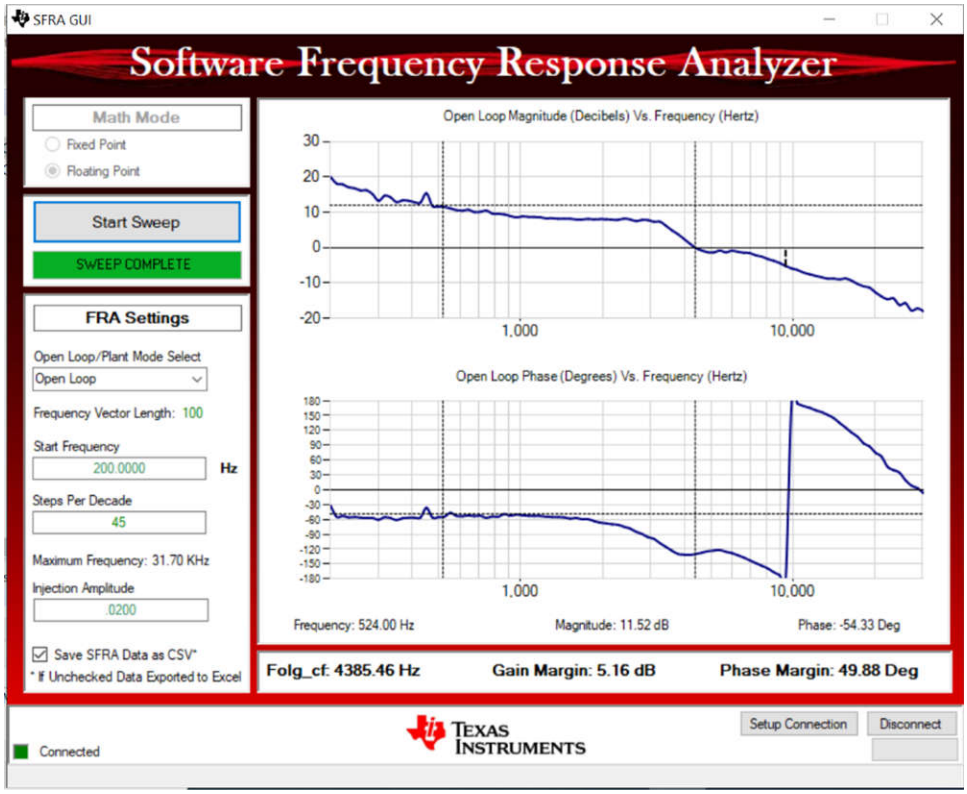



**Figure 3-16. SFRA Run, PFC Closed Current Loop, Open Loop Gain**

7. To bring the system to a safe stop, switch off the output from the AC power supply thus bring the input AC voltage down to zero, observe the TTPLPFC_guiVbus_Volts comes down to zero as well.
8. Fully halting the MCU when in real-time mode is a two-step process. First halt the processor by using the *Halt* button on the toolbar ( ) or by using *Target* → *Halt*. Then take the MCU out of real-time mode by clicking on . Finally, reset the MCU by clicking on .
9. Close CCS debug session by clicking on *Terminate Debug Session* (*Target* → *Terminate all*) .

#### 3.1.2.5.4 Lab 4: Closed Voltage and Current Loop (PFC)

In this lab, the outer voltage loop is closed with the inner current loop closed. A PI-based compensator is used and tuned through the compensation designer for the outer voltage loop. To test lab 2, set the output load to 0.2 A first.

Figure 3-17 shows the software diagram for this lab.

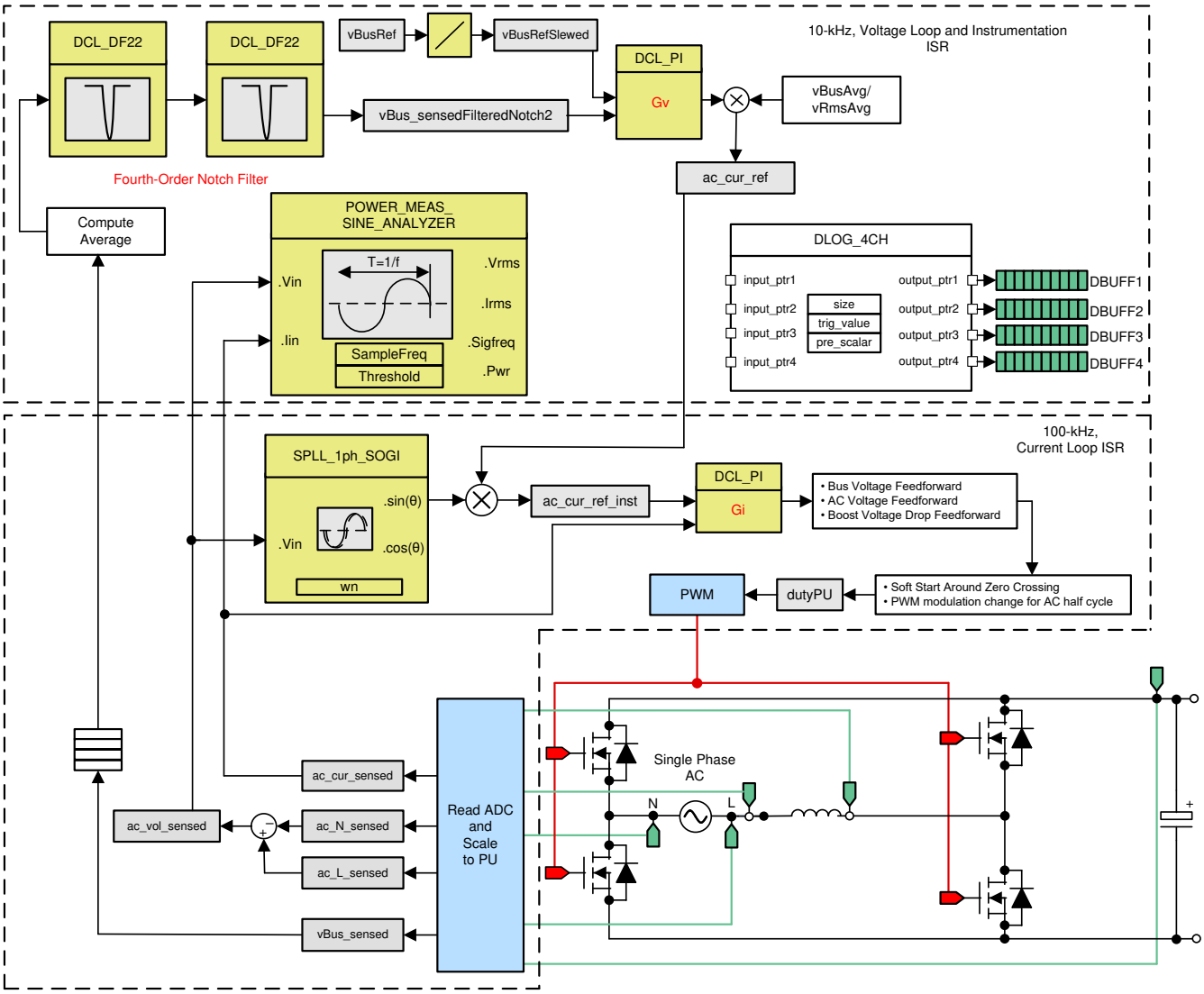



**Figure 3-17. PFC Lab 4 Control Diagram: Output Voltage Control With Inner Current Loop**

##### *3.1.2.5.4.1 Setting Software Options for Lab 4*

1. powerSUITE settings: On the powerSUITE page select under *Project Options* section:
   - Select *Lab 4* under Lab option
   - Also disable the other options such as *Non Linear Voltage Loop, Adaptive Deadtime*
2. Assuming all other options are same as specified earlier in Section 3.1.2.5.1.1.
3. Make sure the load connected at the output of the board is correctly entered on the powerSUITE syscfg page because this load value is used in the design of the voltage compensator.

##### *3.1.2.5.4.2 Building and Loading Project and Setting up Debug*

1. Right click on the project name, and click *Rebuild Project*. The project builds successfully. Click *Run →Debug* or the debug button , which launches a debugging session. In the case of dual CPU devices, a window may appear to select the CPU the debug must be performed. In this case, select CPU1. The

project then loads on the device, and CCS debug view becomes active. The code halts at the start of the main routine.

2. To add the variables in the watch and expressions window, click *View* → *Scripting Console* to open the scripting console dialog box. On the upper-right corner of this console, click on *Open* to browse to the *setupdebugenv_lab4.js* script file located inside the project folder. This file populates the watch window with appropriate variables required to debug the system. Click on the *Continuous Refresh* button ( ) on the watch window to enable continuous update of values from the controller.

   The watch window appears as shown in Figure 3-18.

Variables | Expressions | Registers

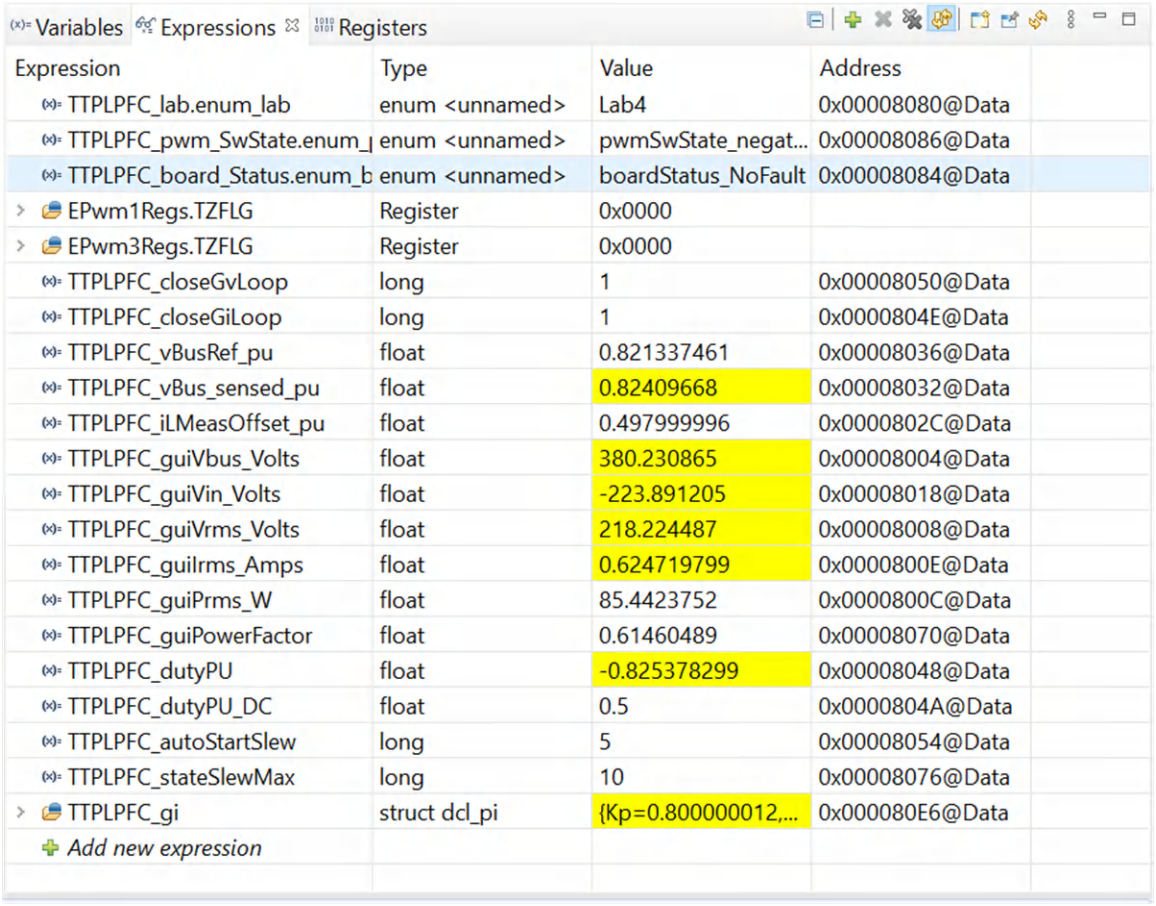
| Expression | Type | Value | Address |
|---|---|---|---|
| TTPLPFC_lab.enum_lab | enum <unnamed> | Lab4 | 0x00008080@Data |
| TTPLPFC_pwm_SwState.enum_ | enum <unnamed> | pwmSwState_defau... | 0x00008086@Data |
| TTPLPFC_board_Status.enum_b | enum <unnamed> | boardStatus_Input... | 0x00008084@Data |
| EPwm1Regs.TZFLG | Register | 0x0004 | |
| EPwm3Regs.TZFLG | Register | 0x0004 | |
| TTPLPFC_closeGvLoop | long | 0 | 0x00008050@Data |
| TTPLPFC_closeGiLoop | long | 0 | 0x0000804E@Data |
| TTPLPFC_vBusRef_pu | float | 0.821337461 | 0x00008036@Data |
| TTPLPFC_vBus_sensed_pu | float | 0.00549316406 | 0x00008032@Data |
| TTPLPFC_iLMeasOffset_pu | float | 0.497999996 | 0x0000802C@Data |
| TTPLPFC_guiVbus_Volts | float | 2.47994113 | 0x00008004@Data |
| TTPLPFC_guiVin_Volts | float | 2.08720708 | 0x00008018@Data |
| TTPLPFC_guiVrms_Volts | float | 0.0 | 0x00008008@Data |
| TTPLPFC_guiIrms_Amps | float | 0.0 | 0x0000800E@Data |
| TTPLPFC_guiPrms_W | float | 0.0 | 0x0000800C@Data |
| TTPLPFC_guiPowerFactor | float | 0.0 | 0x00008070@Data |
| TTPLPFC_dutyPU | float | 0.00999999978 | 0x00008048@Data |
| TTPLPFC_dutyPU_DC | float | 0.5 | 0x0000804A@Data |
| TTPLPFC_autoStartSlew | long | 0 | 0x00008054@Data |
| TTPLPFC_stateSlewMax | long | 10 | 0x00008076@Data |
| TTPLPFC_gi | struct dcl_pi | {Kp=0.800000012,... | 0x000080E6@Data |
| *Add new expression* | | | |

**Figure 3-18. PFC Lab 4: Expressions View**

3. Enable real-time mode by hovering the mouse on the buttons on the horizontal toolbar and clicking the button .
4. Run the project by clicking on .
5. Halt the processor by using the *Halt* button on the toolbar ( ).

#### *3.1.2.5.4.3 Running Code*

1. The project is programmed to wait for input voltage to excel at approximately 75 $V_{RMS}$ to drive the inrush relay, and clear the trip.
2. Run the project by clicking on .
3. Now apply an input voltage of approximately 220 V. The board comes out of the undervoltage condition and inrush relay is driven. The trip clears, and the output rises to 380-V DC. A sinusoidal current is drawn from the AC input. Figure 3-19 shows the watch window when the program is running at this stage.



Variables | Expressions | Registers

| Expression | Type | Value | Address |
|---|---|---|---|
| TTPLPFC_lab.enum_lab | enum <unnamed> | Lab4 | 0x00008080@Data |
| TTPLPFC_pwm_SwState.enum_ | enum <unnamed> | pwmSwState_negat... | 0x00008086@Data |
| TTPLPFC_board_Status.enum_b | enum <unnamed> | boardStatus_NoFault | 0x00008084@Data |
| EPwm1Regs.TZFLG | Register | 0x0000 | |
| EPwm3Regs.TZFLG | Register | 0x0000 | |
| TTPLPFC_closeGvLoop | long | 1 | 0x00008050@Data |
| TTPLPFC_closeGiLoop | long | 1 | 0x0000804E@Data |
| TTPLPFC_vBusRef_pu | float | 0.821337461 | 0x00008036@Data |
| TTPLPFC_vBus_sensed_pu | float | 0.82409668 | 0x00008032@Data |
| TTPLPFC_iLMeasOffset_pu | float | 0.497999996 | 0x0000802C@Data |
| TTPLPFC_guiVbus_Volts | float | 380.230865 | 0x00008004@Data |
| TTPLPFC_guiVin_Volts | float | -223.891205 | 0x00008018@Data |
| TTPLPFC_guiVrms_Volts | float | 218.224487 | 0x00008008@Data |
| TTPLPFC_guiIrms_Amps | float | 0.624719799 | 0x0000800E@Data |
| TTPLPFC_guiPrms_W | float | 85.4423752 | 0x0000800C@Data |
| TTPLPFC_guiPowerFactor | float | 0.61460489 | 0x00008070@Data |
| TTPLPFC_dutyPU | float | -0.825378299 | 0x00008048@Data |
| TTPLPFC_dutyPU_DC | float | 0.5 | 0x0000804A@Data |
| TTPLPFC_autoStartSlew | long | 5 | 0x00008054@Data |
| TTPLPFC_stateSlewMax | long | 10 | 0x00008076@Data |
| TTPLPFC_gi | struct dcl_pi | {Kp=0.800000012,... | 0x000080E6@Data |
| *Add new expression* | | | |

**Figure 3-19. PFC Lab 4: Expressions View After AC Voltage is Applied**

4. SFRA is integrated in the software of this build to verify the designed compensator provides enough gain and phase margin by measuring on hardware. To run the SFRA, keep the project running, and from the syscfg page, click on the *SFRA* icon. SFRA GUI appears.
5. Select the options for the device on the SFRA GUI. For example, for F28004x, select floating point. Click on *Setup Connection*, and on the pop-up window, uncheck the boot on connect option and select an appropriate COM port. Click *OK*. Return to the SFRA GUI, and click *Connect*.
6. The SFRA GUI connects to the device. A SFRA sweep can now be started by clicking *Start Sweep*. The complete SFRA sweep takes a few minutes to finish. Activity can be monitored by seeing the progress bar on the SFRA GUI and checking the flashing of blue LED on the back on the control card that indicates UART activity. Once complete, a graph with the open loop plot appears, as seen in Figure 3-20. This action verifies that the designed compensator is indeed stable. The frequency response data is also saved in the project folder under an SFRA data folder and is time stamped with the time of the SFRA run.

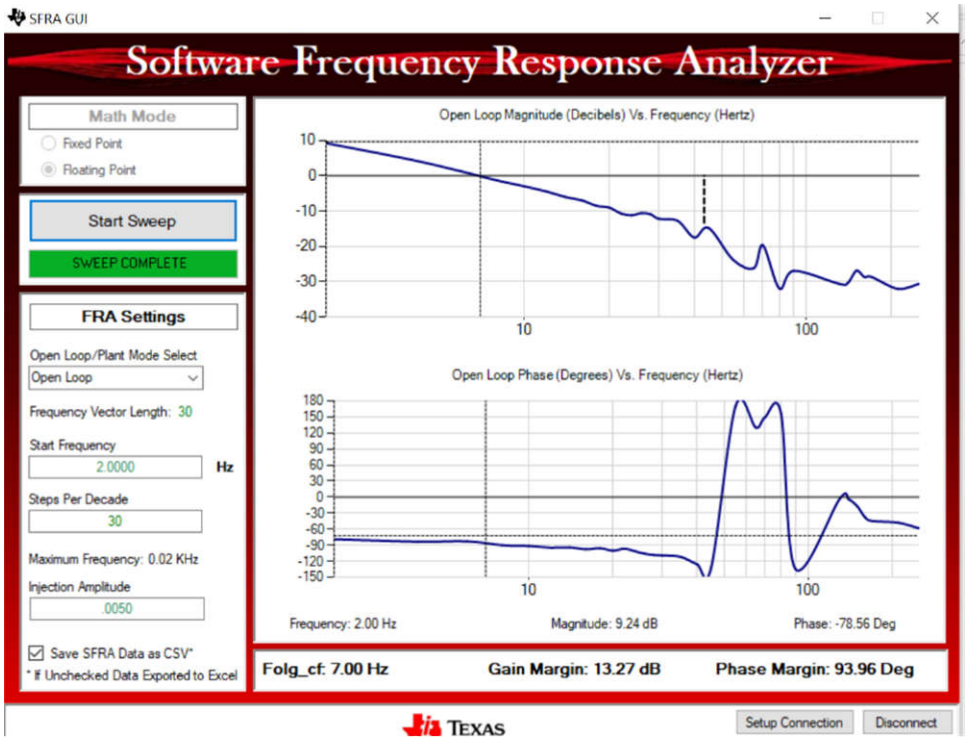



**Figure 3-20. SFRA Run on PFC Closed Voltage Loop**

7. This verifies the voltage compensator design.
8. To bring the system to a safe stop, switch off the output from the AC power supply thus bring the input AC voltage down to zero, observe the TTPLPFC_guiVbus_Volts comes down to zero as well.
9. Fully halting the MCU when in real-time mode is a two-step process. First halt the processor by using the *Halt* button on the toolbar ( ) or by using *Target* → *Halt*. Then take the MCU out of real-time mode by clicking on . Finally, reset the MCU by clicking on .
10. Close CCS debug session by clicking on *Terminate Debug Session* (*Target* → *Terminate all*) .

### 3.1.3 LLC Stage Software

#### *3.1.3.1 Opening Project Inside CCS*

1. Open CCS (version 10.1, or newer). Maximize CCS to fill the screen. Close the welcome screen if it opens up.
2. On the menu bar click **Project** > **Import CCS Project**. Below the root directory, navigate to and select… *\C2000Ware_DigitalPower_SDK_X_XX_XX_XX\solutions\tida_010062* directory. Make sure that below the **Projects** tab, llc_F28002x, llc_F28003x, or llc_F28004x is selected. Click **Finish**.
3. LLC project should now appear in the CCS Project Explorer window. This project will invoke all the necessary tools (compiler, assembler, and linker) to build the project. A project contains all the files and build options required to develop an executable output file (.out), which can be run on the MCU hardware.
4. In the project window on the left, click the arrow sign to the left of the project name. The project windows will look like Figure 3-21.

### 3.1.3.2 Project Structure

Once the project is imported, the project explorer appears inside CCS as shown in Figure 3-21.

For this stage of this design <solution> is *llc*. The project structure is the same with ttplpfc in Section 3.1.2.2.

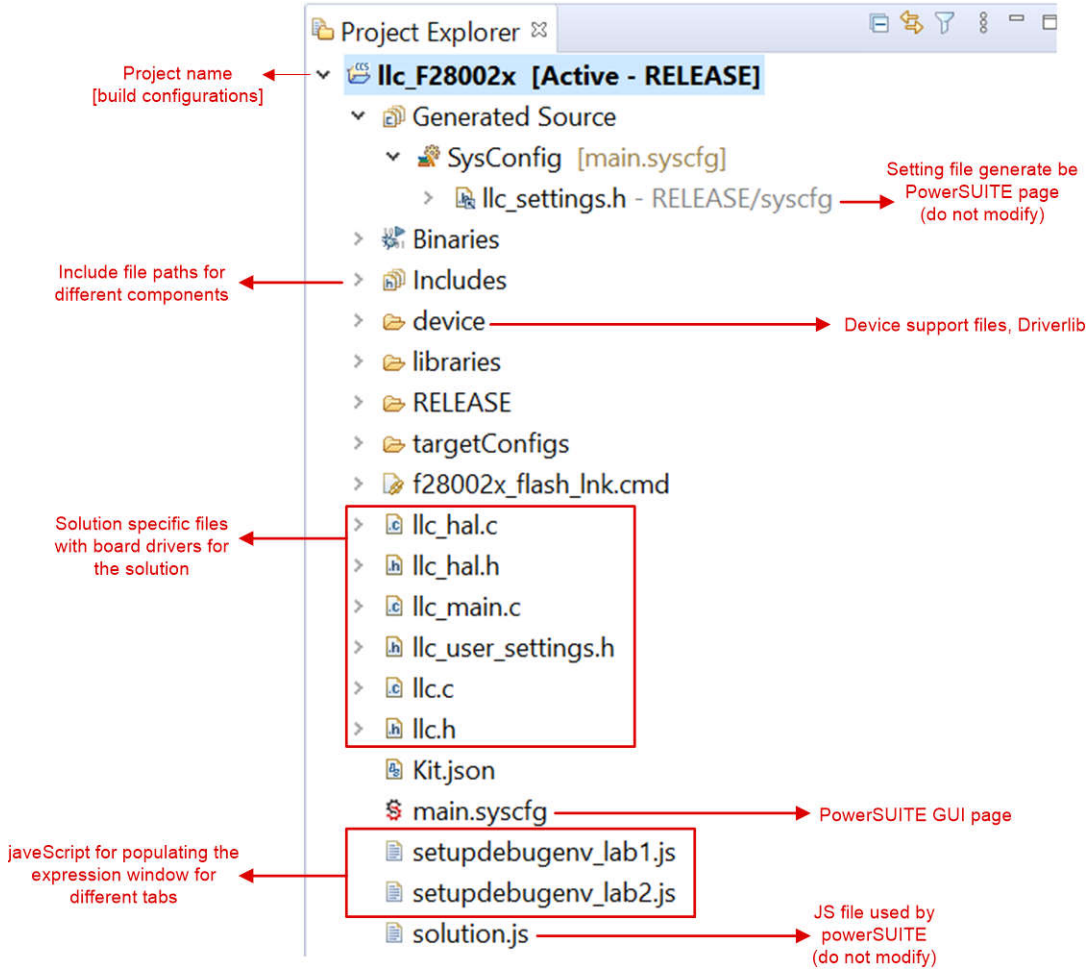



**Figure 3-21. Project Explorer View of LLC Solution Project**

Double click the main.sysconfig file name in the project window. This GUI can be used to change the parameters for an adapted solution, like power rating, inductance, capacitance, sensing circuit parameters, and so forth.

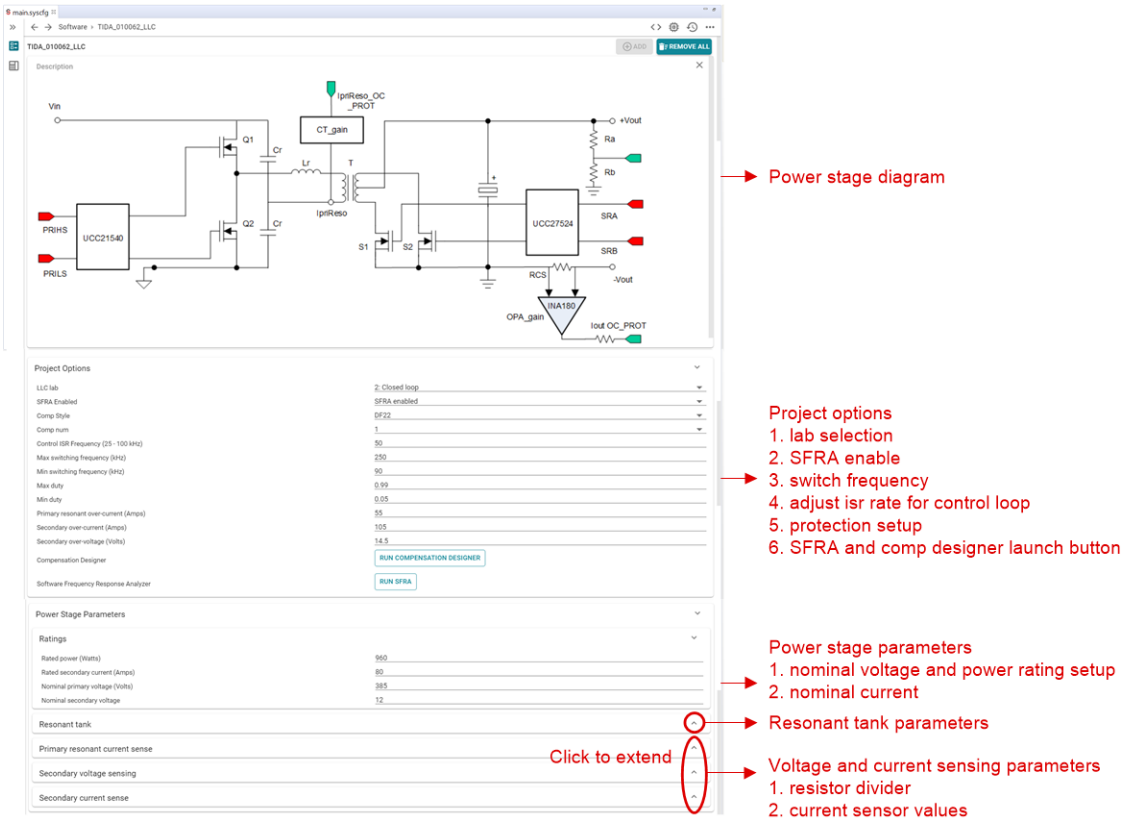



**Figure 3-22. powerSUITE Page for LLC Stage**

### *3.1.3.3 Software Flow*

The software project makes use of the C-background and C-ISR framework. The project uses C background code as the main supporting program for the application, which is responsible for all system management tasks, decision making, intelligence, and host interaction. The C-ISR code is executed inside a time critical interrupt service routine (ISR), and runs all the critical control code. This code includes ADC reading, control calculations, and PWM updates. The *Control ISR* portion of the C-ISR is executed at a fixed rate of 50 kHz using a spare PWM module timer. Figure 3-23 shows the software flow diagram of the firmware.

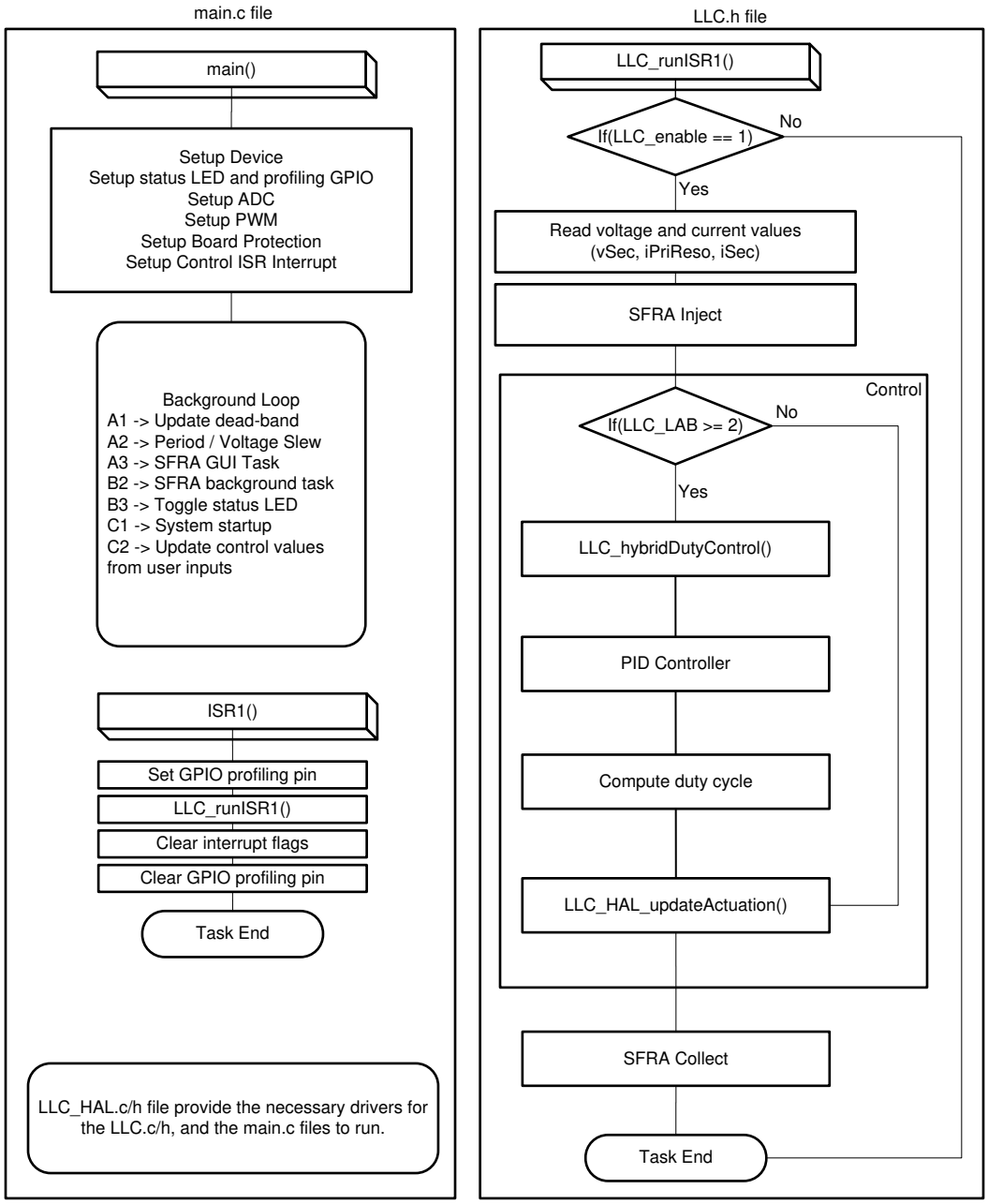



**Figure 3-23. LLC Stage Software Structure**

This project is divided into two incremental labs, which makes learning and getting familiar with the board and software easier. This approach is also good for debugging and testing boards.

Lab 1: Open-Loop Control

Lab 2: Closed-Loop Control With SFRA

These labs are detailed in Section 3.1.3.5. If using the reference design hardware, make sure the hardware setup is completed as outlined in Section 3.1.1.

### *3.1.3.4 CPU Utilization and Memory Allocation*

ISR loadings with advanced options enabled can be measured and shown as follows.

| | ISR1 (50 kHz) | ISR2 (10 kHz) |
|---|---|---|
| CPU utilization | 22% | -- |

The memory allocation (F28002x) is shown in Figure 3-24.


**Figure 3-24. TIDA-010062 LLC Stage Memory Allocation (F28002x)**

### *3.1.3.5 Running the Project*

#### 3.1.3.5.1 Lab 1: Open-Loop Control

The objective of this lab is to get familiar with the TIDA-010062 LLC stage hardware and control the output voltage using direct PWM period adjustments without feedback. Because this system is running open loop, the ADC measured values are only used for instrumentation purposes in this lab. The PWM period is adjusted using the Expressions Window. In this way, this lab verifies the sensing of feedback values from the power stage and also operation of the PWM gate driver and ensures there are no hardware issues. Additionally, calibration of input and output voltage sensing can be performed in this lab.

Figure 3-25 shows the software block diagram for LLC lab 1.

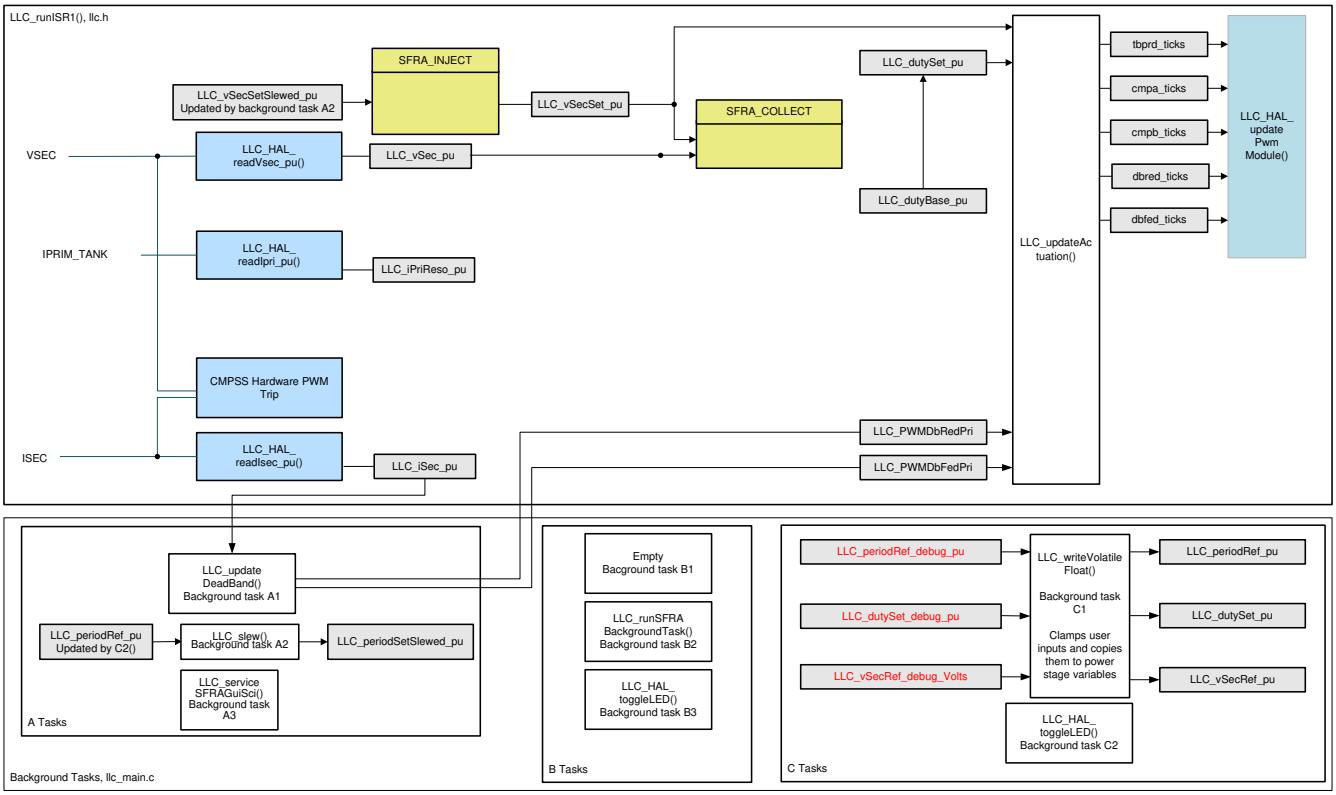



**Figure 3-25. LLC Lab 1 Software Diagram**

Before tests with lab 1, make sure the hardware setup is completed as outlined in Section 3.1.1.3.

#### *3.1.3.5.1.1 Software Setup*

1. powerSUITE Settings: Select Lab1 under the Lab option / Project Options section.
2. If this is an adapted solution, edit the setting under *Voltage and Current Sensing Parameters*.
3. Under Power Stage Parameters specify the switching frequency, the dead band, and the power rating.
4. Under Resonant Tank specify the resonant tank parameters.
5. Save the page.

#### *3.1.3.5.1.2 Build and Load the Project*

1. Right-click on the project name, and click on **Rebuild Project**.
2. Click the **Debug** button ( ) or click **Run** > **Debug**. The code should compile and load.
3. The program should be stopped at the start of **main()**.

#### *3.1.3.5.1.3 Debug Environment Windows*

It is standard debug practice to watch local and global variables while debugging code. There are various methods for doing this in CCS, such as memory windows and watch windows. Additionally, CCS has the ability to make time (and frequency) domain plots, which allows the user to view waveforms using graph windows.

Populate the Expressions Window entries by clicking **View** > **Scripting console** on the menu bar and then opening the *setupdebugenv_lab1.js* file from the project directory using the scripting console **Open File** ( ) command. The Expressions Window should look like Figure 3-26.

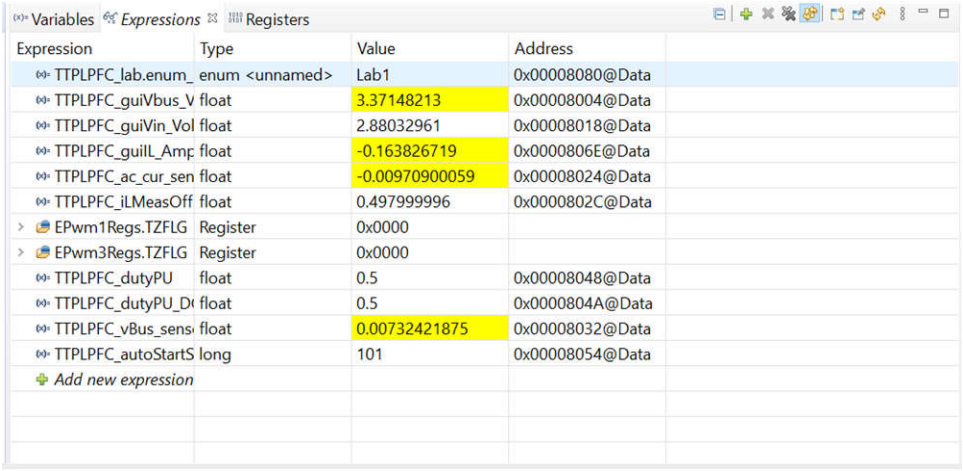



**Figure 3-26. LLC Lab 1 Expressions View**

**Table 3-3. LLC Lab 1 Description of Expressions Window Entries**

| VARIABLE | DESCRIPTION |
|---|---|
| LLC_buildLevel | Shows which lab is loaded. |
| LLC_board_Status | Show the board status: Fault types or normal operation |
| LLC_startFlag | Set this variable to 1 to start the powerstage. |
| LLC_vSec_Volts | Output voltage in Volts. |
| LLC_iPriReso_Amps | The primary tank current in Amps. |
| LLC_iSec_Amps | Secondary (output) current in Amps. |
| LLC_periodRef_debug_pu | The variable used to control the switching period in lab 1. It is in per unit format. |
| LLC_dutySet_debug_pu | The variable used to control the switching duty It is in per unit format. |
| EPwm1Regs.CMPA.CMPA | Compare A register in ePWM1. |
| EPwm1Regs.CMPB.CMPB | Compare B register in ePWM1. |
| EPwm1Regs.TBPRD | Time-based period register in ePWM1. |
| EPwm2Regs.CMPA.CMPA | Compare A register in ePWM2. |
| EPwm2Regs.CMPB.CMPB | Compare B register in ePWM2. |
| EPwm2Regs.TBPRD | Time-based period register in ePWM2. |
| EPwm1Regs.TZFLG | TZ flag of ePWM1. For debugging protection use. |
| EPwm2Regs.TZFLG | TZ flag of ePWM2. For debugging protection use. |

#### *3.1.3.5.1.4 Run the Code*

1. Run the code by using the <F8> key or the Run button ( ) on the toolbar
2. As this is an open-loop test (no voltage loop), care must be taken not to use too small a load to avoid accidental high output voltages. Use a load of 10 A at the 12-V output.
3. In the Expressions Window set LLC_startFlag = 1. The converter operation should start with the default LLC_periodSet_pu value resulting in approximately 250-kHz switching frequency.
4. The LLC_periodRef_debug_pu value may be changed between 0.55 and 0.65. As this value is increased the switching frequency decreases, which results in a higher energy delivered to the load. In open loop this results in an increase in output voltage, which should not be allowed to exceed board capabilities.
5. Set input voltage: increase DC source output from 0 V to 380 V. Verify that all of the sensed values are updating in the expressions window: LLC_vSec_Volts, LLC_iPriReso_Amps, LLC_iSec_Amps.

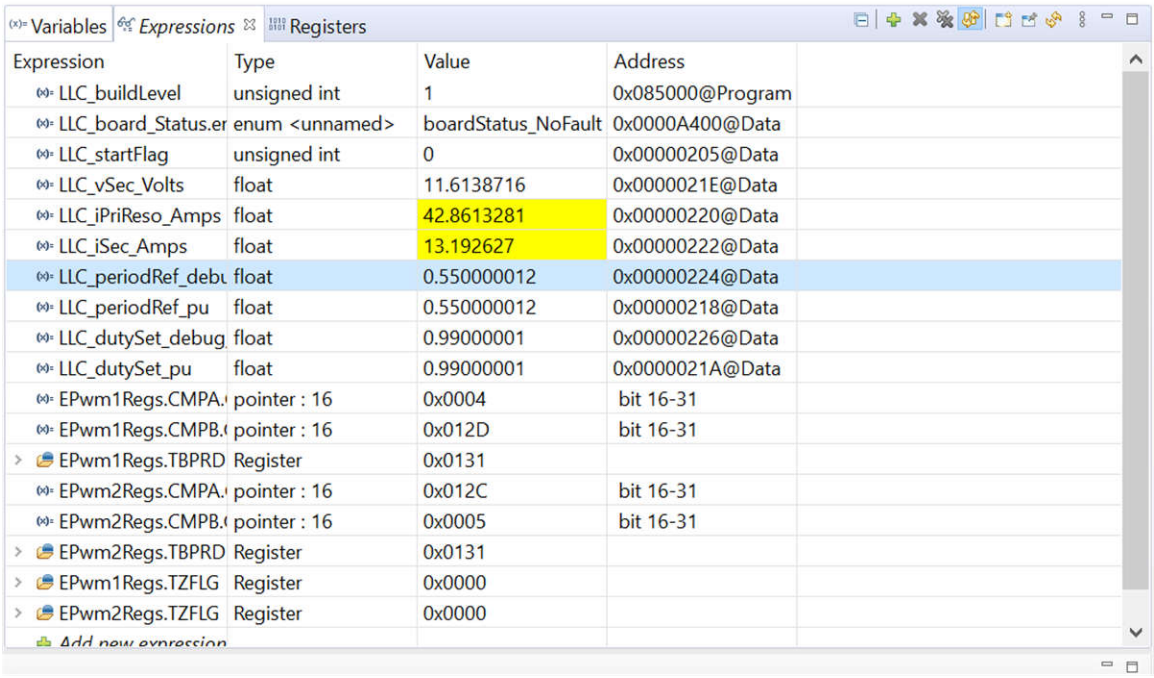



**Figure 3-27. LLC Lab 1: Watch Expression, After Open Loop Operation Begins**

6. Turn OFF the 380-V DC power supply.
7. Fully halting the MCU when in real-time mode is a two-step process. First halt the processor by using the *Halt* button on the toolbar ( ) or by using *Target* → *Halt*. Then take the MCU out of real-time mode by clicking on . Finally, reset the MCU by clicking on
8. Close CCS debug session by clicking on *Terminate Debug Session* (*Target* → *Terminate all*) .

#### 3.1.3.5.2 Lab 2: Closed-Loop Control With SFRA

The objective of this lab is to regulate the output voltage of the converter using closed-loop feedback control realized in the form of a software-coded control loop. The SFRA GUI can be used during run time to capture the frequency response of the system.

Figure 3-28 shows the software block diagram for lab 2.

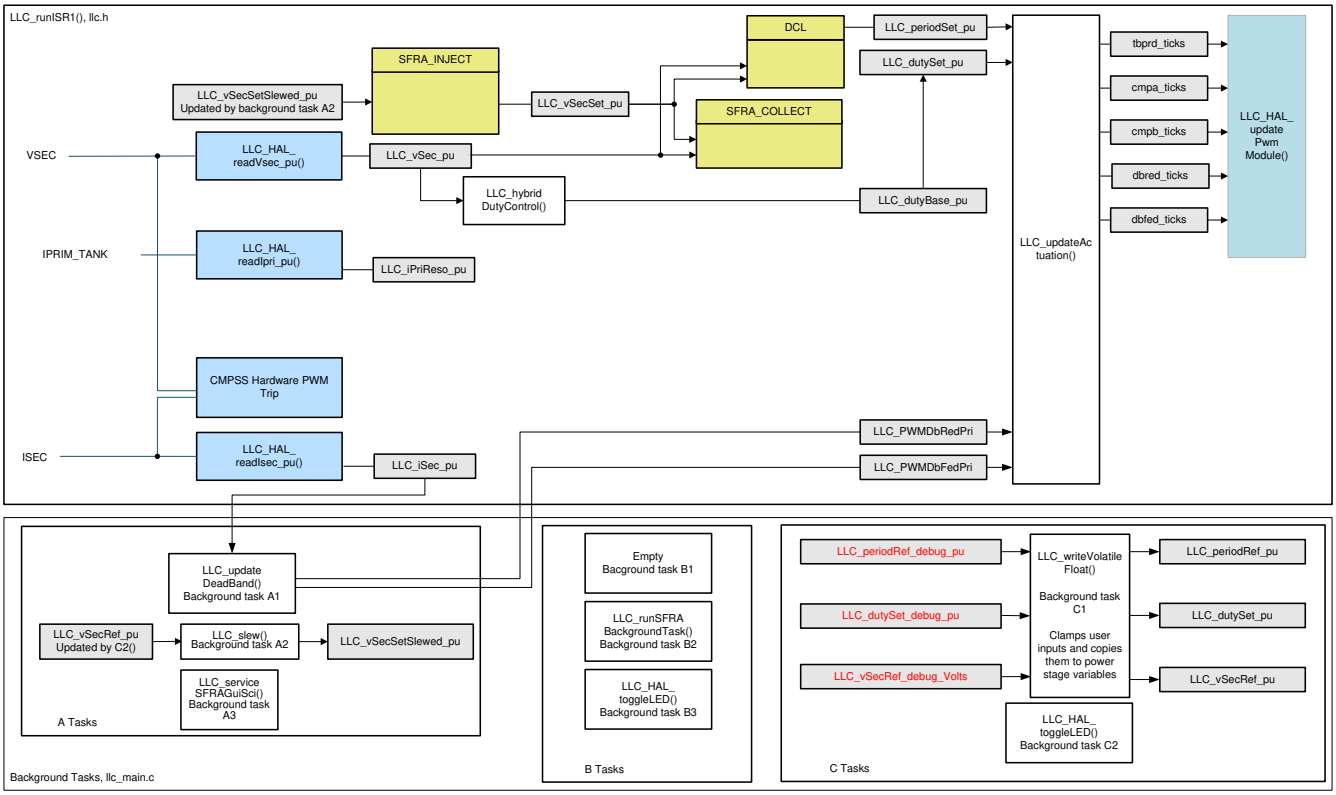



**Figure 3-28. LLC Lab 2 Software Diagram**

Before the test of lab 2, make sure the hardware setup is completed as outlined in Section 3.1.1.3.

##### *3.1.3.5.2.1 Software Setup*

1. powerSUITE Settings: Select Lab2 under the Lab option / Project Options section
2. Assuming all other options are same as specified in Section 3.1.3.5.1.1
3. Save the page

##### *3.1.3.5.2.2 Build and Load the Project*

Follow the steps in Section 3.1.3.5.1.2 for the lab one procedure.

##### *3.1.3.5.2.3 Debug Environment Windows*

Populate the Expressions Window entries by clicking on *View* → *Scripting console* on the menu bar and then opening the *setupdebugenv_lab2.js* file from the project directory using the scripting console *Open File* (📂) command. The Expressions Window should look like Figure 3-29.

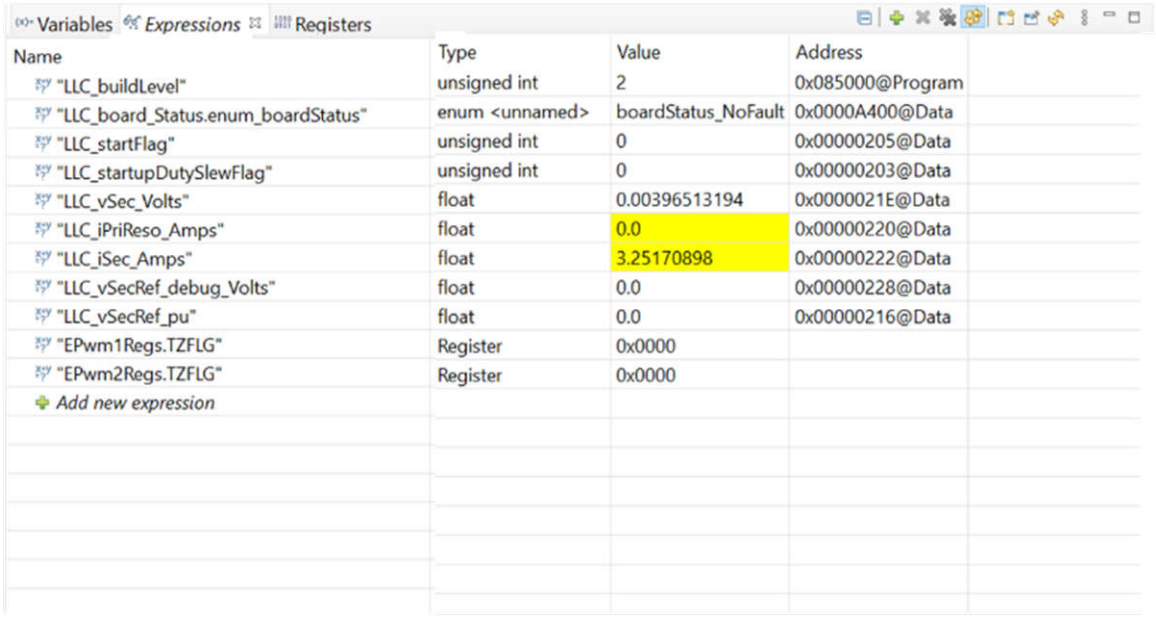



**Figure 3-29. LLC Lab 2 Expressions View**

**Table 3-4. LLC Lab 2 Description of Expressions Window Entries**

| VARIABLE | DESCRIPTION |
|---|---|
| LLC_buildLevel | Shows which lab is loaded. |
| LLC_board_Status | Show the board status: Fault types or normal operation |
| LLC_startFlag | Set this variable to 1 to start the powerstage. |
| LLC_startupDutySlewFlag | Start up flag: duty slew status. |
| LLC_vSec_Volts | Output voltage in Volts. |
| LLC_iPriReso_Amps | The primary tank current in Amps. |
| LLC_iSec_Amps | Secondary (output) current in Amps. |
| LLC_vSecRef_debug_Volts | This variable allows the user to set the desired regulation voltage. |
| LLC_vSecRef_pu | Output voltage reference in pu. |
| EPwm1Regs.TZFLG | TZ flag of ePWM1. For debugging protection use. |
| EPwm2Regs.TZFLG | TZ flag of ePWM2. For debugging protection use. |

#### *3.1.3.5.2.4 Run the Code*

1. Run the code by using the <F8> key or the *Run* button ( ) on the toolbar.
2. Use a load of 10 A at the 12-V output.
3. Set the 380-V DC supply. Set the power supply current limit to an appropriate level for this test. Next, turn ON this 380-V power supply.
4. At this point the output voltage should still be zero as the converter *Start* command has not been initiated.
5. Now set the *LLC_startFlag* to *1* in the *Expressions Window*.
6. The converter operation should start and the output should ramp-up to approximately 12 V.

**Note**

If the output voltage does not ramp up to approximately 12 V, turn OFF the 380-V DC supply immediately. Verify lab 1 operation first as described in Section 3.1.3.5.1. The user may also be required to re-verify the board components and debug hardware issues (components do not match the bill of materials (BOM), PCB fabrication issue, and so forth) before this board can be tested again.

7. Observe the effect of varying load on the output voltage and input current. There should be virtually no effect on the output voltage. Similarly, observe the effect of varying the input voltage. Again there should be virtually no effect on the output voltage.

8. Different waveforms, like the PWM gate drive signals, input voltage, and current and output voltage may also be probed using an oscilloscope. Appropriate safety precautions should be taken and appropriate grounding requirements should be considered while probing these high voltages and high currents for this isolated DC/DC converter.
9. Select the main.syscfg file and open SFRA. Click on setup connection, and select the appropriate COM port. Make sure that the baud rate is set to *57600* and that *Boot on Connect* is unchecked. Click the *OK* button. Select *Floating Point* math. Click *Connect* on the SFRA GUI. Once the GUI is connected, click on *Start Sweep*. SFRA will start applying different frequencies and collecting the response for frequency analysis.

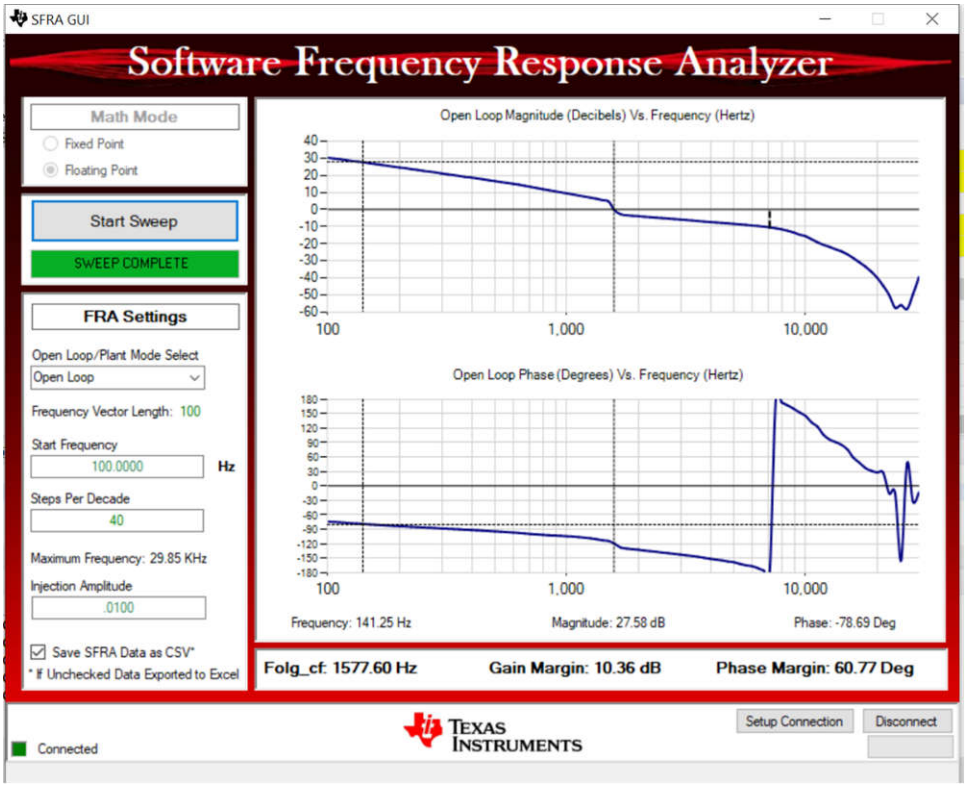



**Figure 3-30. SFRA Run, LLC Closed Loop, Open Loop Gain**

10. Turn OFF the 380-V DC power supply.
11. Fully halting the MCU when in real-time mode is a two-step process. First halt the processor by using the *Halt* button on the toolbar ( ) or by using *Target* → *Halt*. Then take the MCU out of real-time mode by clicking on . Finally, reset the MCU by clicking on .
12. Close CCS debug session by clicking on *Terminate Debug Session* (*Target* → *Terminate all*) .

### 3.1.4 PFC + LLC Stage Dual Test

#### *3.1.4.1 Hardware Setup*

For f28004x and f28002x based solutions, no extra hardware is required to conduct PFC + LLC two-stage test. To test the f28003x-based solution, two FSI adapter boards (TMDSFSIADAPEVM) are required as shown in Figure 3-31. The TMDSFSIADAPEVM is an evaluation board that assists in understanding the functionality of the FSI communications peripheral of the C2000. Two FSI adapter boards are connected to the J1 header in f28003x controlCARDs through the ribbon cable. Due to the limited space, one or two extra connectors are possibly needed in Figure 3-32.

Each FSI core has three signals associated with it: one clock and two data signals. Data is always transmitted or received with the most significant bit of each frame field being first. If multi-lane transmissions are not used, the TXD1 and RXD1 signals can be left unconnected and their GPIOs repurposed for other application needs. In the TIDA-010062, only TXD0 and RXD0 signals are used.

The FSI clock is set to 50 MHz for f28004x and f28002x, and it is set to 60 MHz for f28003x. Customers have the options to lower the clock speed if the noise is significant and impacts the FSI connection.

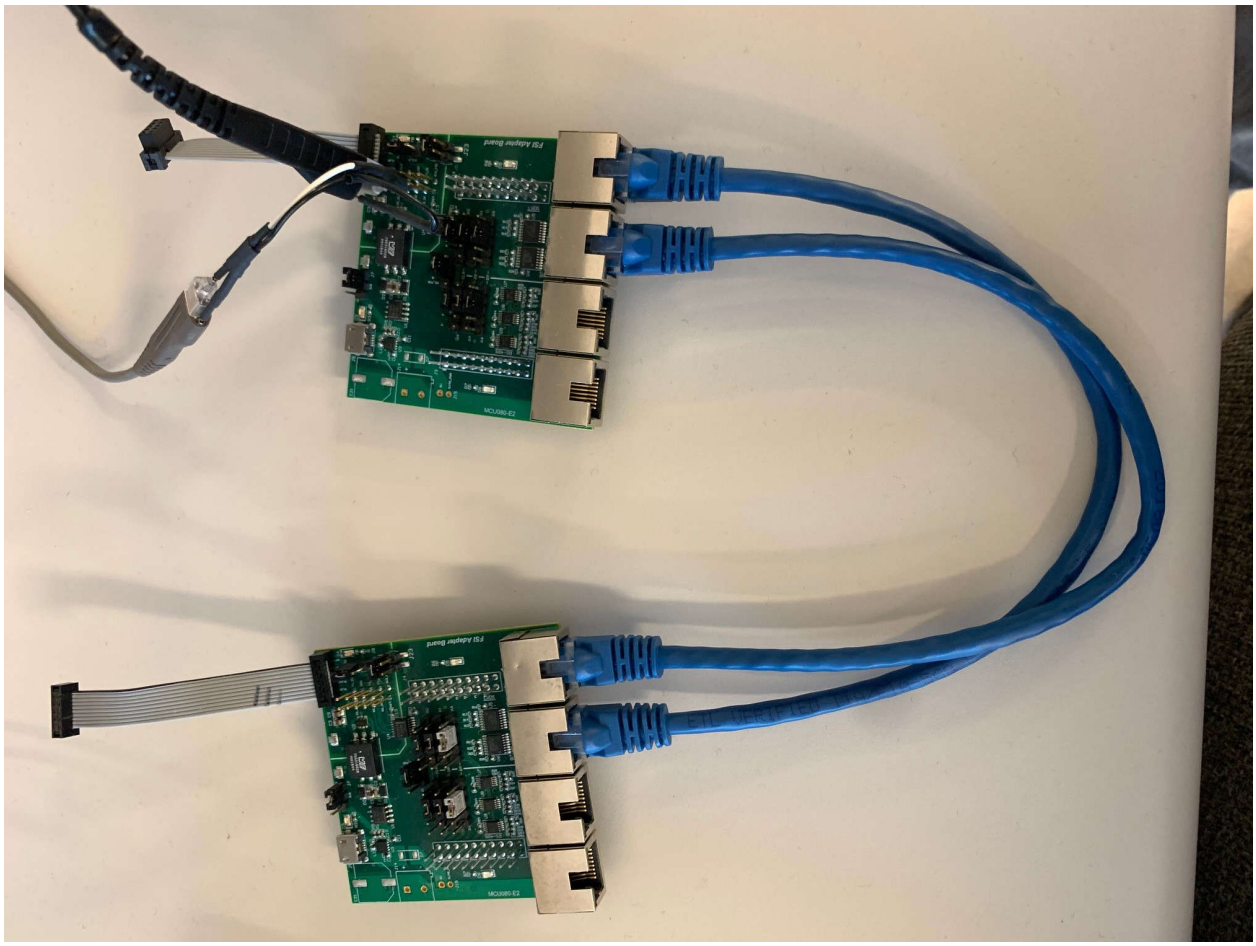


**Figure 3-31. FSI Adapter Boards for f28003x-Based Solution**

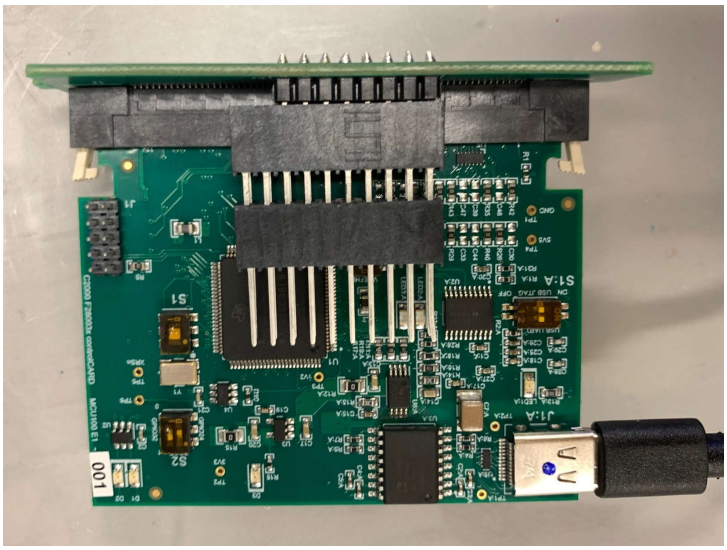


**Figure 3-32. Connector With f28003x Control Card**

### *3.1.4.2 System Test Procedure*

1. Connect the GaN daughter cards and C2000 controlCARD to the mother board. The details of the connections to each pin of the connectors are found in the schematic of the 1-kW, 80+ Titanium AC/DC Power Supply Reference Design.
   - One GaN daughter card *LMG3410-HB-EVM* at connectors Brd1 position
   - C2000 Piccolo controlCARD with *TMDSCNCD280049C*, *TMDSCNCD280039C*, or *TMDSCNCD280025C* at the connector J1 and J3
2. Connect the 3 pins AC terminal of the reference board to the AC power source
3. Connect output terminals with wires to the electronic load, and maintain correct polarity
4. Put a cooling fan at the AC side to maintain airflow on the board
5. Set and maintain a small load of approximately 10 A
6. Connect current and voltage probes to observe the input current, input voltage, and output voltages

### *3.1.4.3 FSI Software in TIDA-010062*

The FSI is a serial peripheral capable of reliable high-speed communication across isolation barriers.

1. Make sure TTPLPFC_LAB in PFC solution is set to 4 and LLC_AUTOSTART in LLC solution is set to 1.
2. The primary side (PFC) MCU and secondary side (LLC) MCU will continue to handshake until both MCUs recognize each other. Once the handshake is completed, TTPLPFC_Handshake_Complete_Flag should be 1. This is an inside background task. Check FsiRxARegs.RX_EVT_STS registers in the primary and secondary side MCU if the flag is not 1.
3. When the PFC output voltage reaches 380 V, the primary MCU will automatically send a flush sequence together with a ping frame to the secondary side MCU to start LLC. For more details of flush sequence and ping frame, see the *TMS320F28004x Real-Time Microcontrollers Technical Reference Manual*, *TMS320F28003x Real-Time Microcontrollers Technical Reference Manual*, or *TMS320F28002x Real-Time Microcontrollers Technical Reference Manual*.

Figure 3-33 shows the clock signal and data signal of flush sequence and ping frame waveform.

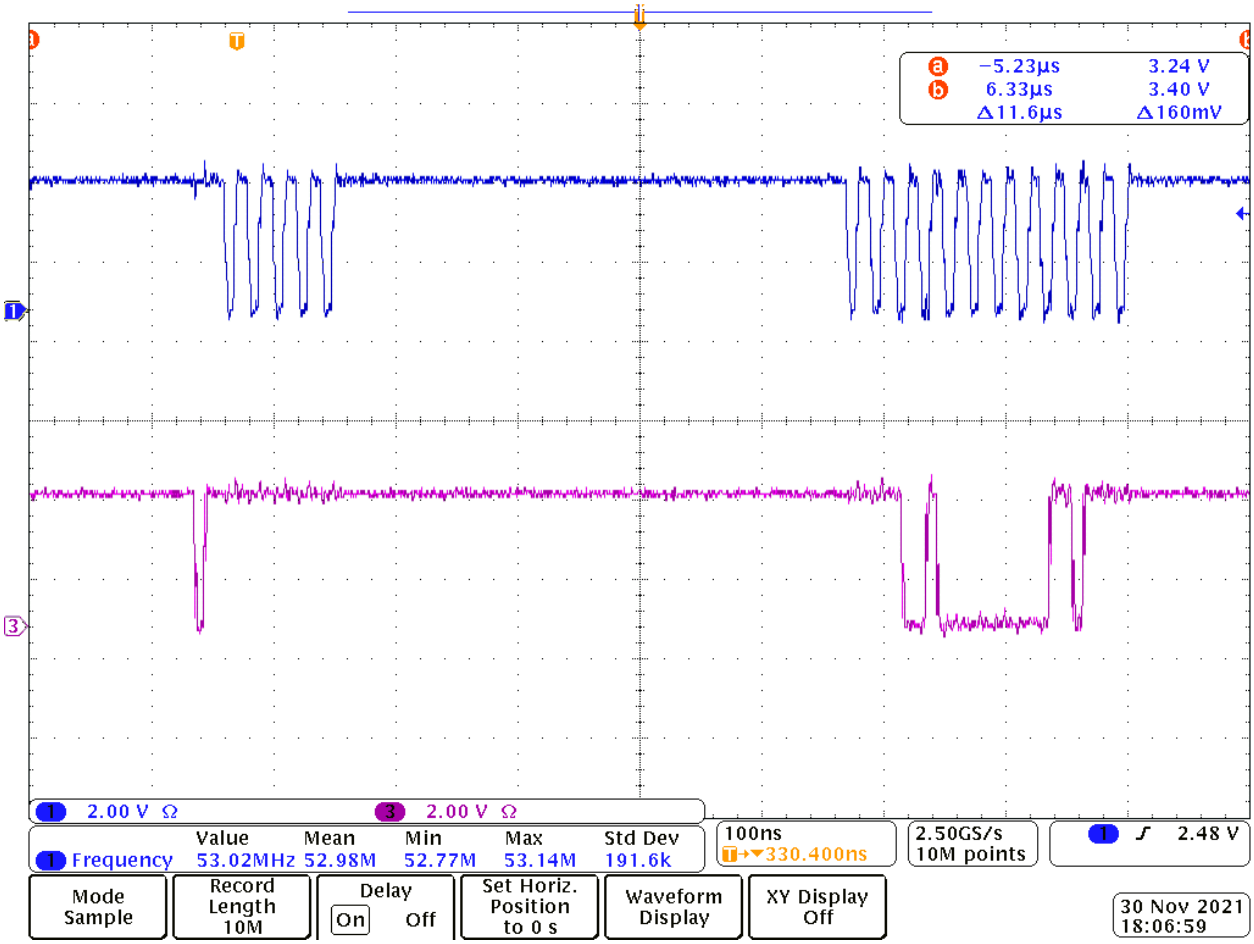



**Figure 3-33. Clock Signal (CH1) and Data Signal (CH3) of Flush Sequence and Ping Frame**

Figure 3-34 shows the clock signal and data signal of ping frame waveform.

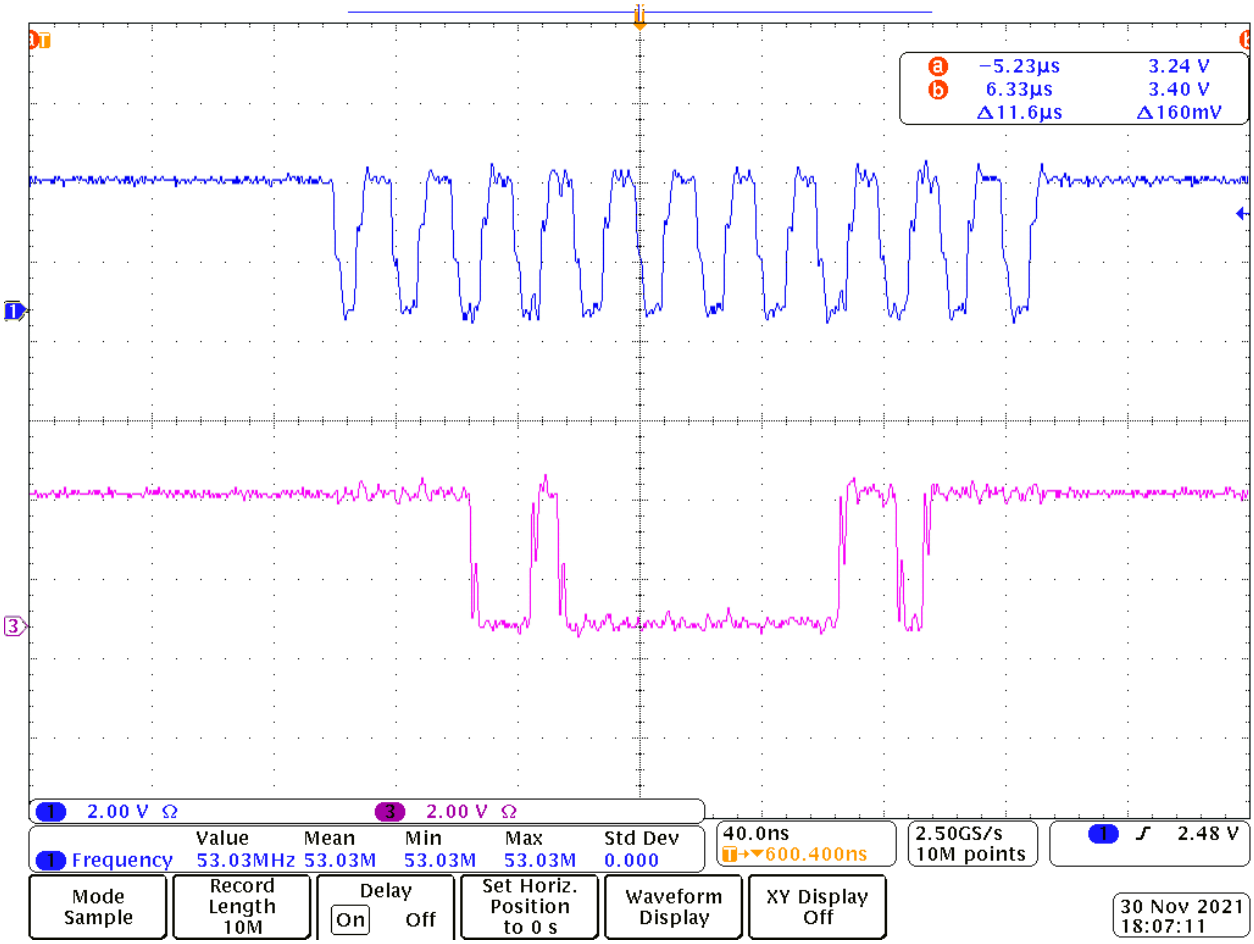



**Figure 3-34. Clock Signal (CH1) and Data Signal (CH3) of Ping Frame**

### 3.1.5 Live Firmware Update Overview

#### *3.1.5.1 Live Firmware Update Description*

In applications such as server power supply, metering, and so on, systems are designed to be run continuously to reduce downtime during firmware updates. But typically, during firmware upgrades, due to bug fixes, new features, and or performance improvements, the system is removed from service causing downtime for associated entities as well. This can be handled with redundant modules but with an increase in total system cost. An alternative approach is to perform a Live Firmware Update (LFU), which allows updating the firmware while the system is still operating. Switching to new firmware can be done either with or without resetting the device, with the latter being more complex.

Section 3.1.5.2 to Section 3.1.5.5 describe how Live Firmware Update (LFU) is implemented in this reference design. The example details LFU without Device Reset using two Flash banks on a TMS320F28003x. The example illustrates LFU capabilities with the main control loop running on the CLA and background processes running on the C28x CPU.

For more detailed information on the specifics of LFU without device reset implementation on C2000™ devices, see the Live Firmware Update Reference Design with C2000™ Real-Time MCUs design guide.

Figure 3-35 shows the block diagram of a typical LFU-based system.

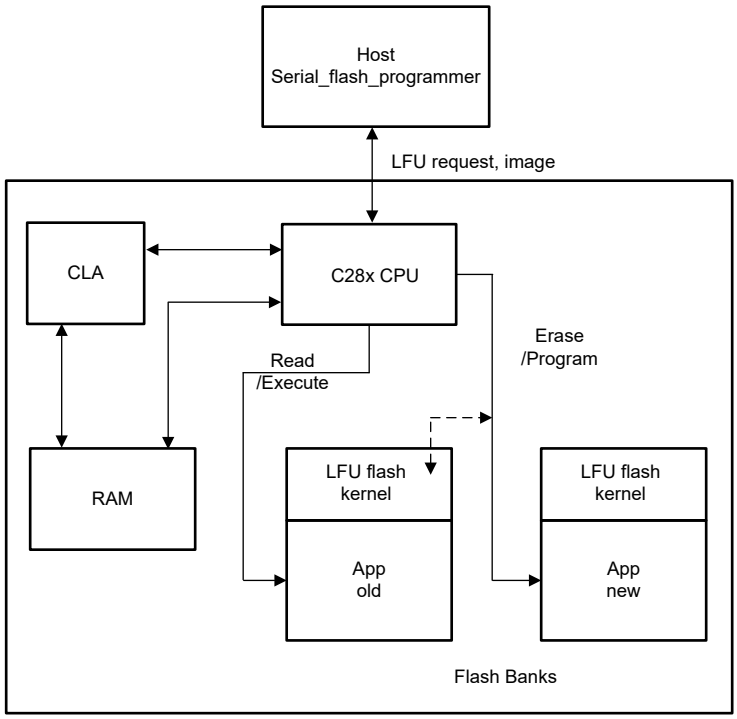



**Figure 3-35. Typical LFU-Based System**

### *3.1.5.2 Software Structure*

Figure 3-36 shows the software directory structure for the LFU concept for F28003x.

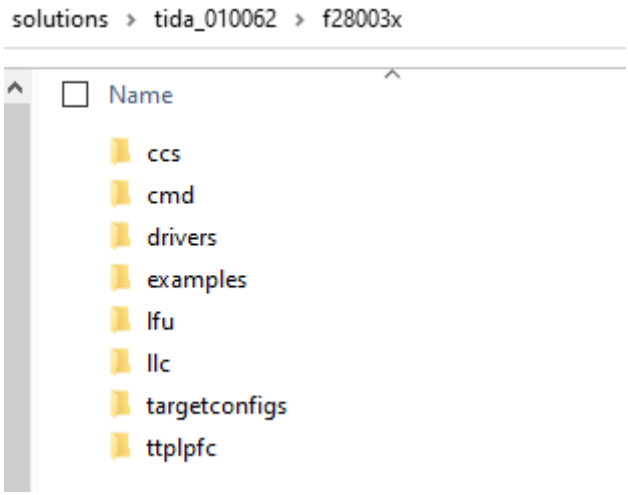



**Figure 3-36. LFU Software Directory Structure**

The directories shown contain files that demonstrate the LFU implementation on the TIDA-010062 design. The following folders are present:

- `/lfu` contains source and header files specific to LFU
- `/drivers` contains HAL (hardware abstraction layer) source and header files
- `/ccs` contains LFU CCS project specifications
- `/cmd` contains linker command files
- `/llc` contains llc_main.c, llc_clatasks.cla, main.syscfg, and other header files
- `/examples` contains the custom bootloader (SCI Flash kernel)

### 3.1.5.3 LFU on LLC Stage Software

#### 3.1.5.3.1 Opening Project Inside CCS

Use the following steps to start the project in CCS:

1. Install CCS from the Code Composer Studio (CCS) Integrated Development Environment (IDE) tools folder, CCSV12.0 or above is recommended
2. Install C2000Ware DigitalPower SDK at the C2000Ware Digital Power SDK tools folder.

**Note**

powerSUITE is installed with the SDK in the default install.

3. Go to *View* → *Resource Explorer*. Under the *TI Resource Explorer*, go to *C2000Ware DigitalPower SDK*. To open the reference design software as it is (opens firmware as it was run on this design and hardware, requires the board to be exactly the same as this reference design)
4. Under C2000Ware DigitalPower SDK, select Development Kits → CCM Totem Pole PFC TIDA-010062, and click on *Run <Import> Project*
5. These steps import the project, and the development kit or designs page shows up. This page can be used to browse all the information on the design including this user guide, test reports, hardware design files, and so forth.
6. Click *Import F28003x LFU DC-DC Project*
7. This action imports the project into the workspace environment, and a main.syscfg page with a GUI similar to Figure 3-37 appears

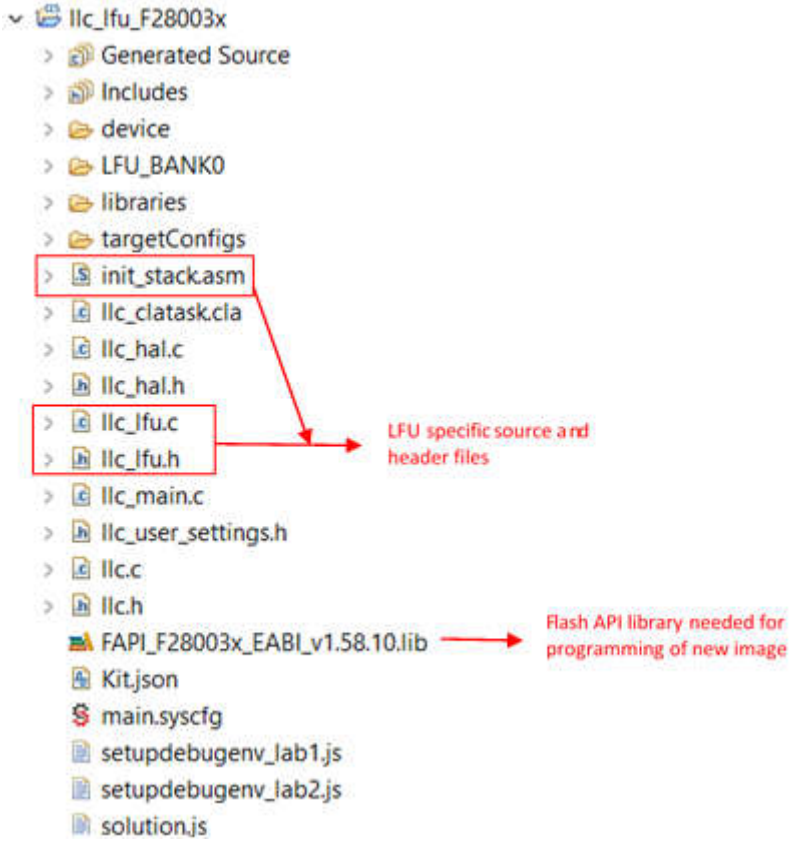



**Figure 3-37. Typical LFU-Based System**

Click the `main.syscfg` file, a GUI page opens with modifiable options for the PFC design. Use this GUI to change the parameters for an adapted concept, like power rating, inductance, capacitance, sensing circuit parameters, and so forth.

For the purposes of LFU, make sure the following options are selected:

- LLC Lab → Closed Loop

- Control On → CLA
- SFRA Enabled → Disabled

8. Once the specified settings have been selected. Build the project for each different build configuration in this order: LFU_BANK0, LFU_BANK1.

**Note**

The project needs to be built in this order since the firmware on Bank1 is built using Bank0 as a reference.

See Section 3.1.3.5 for more information on the specific labs on how to debug them on hardware.

#### *3.1.5.4 Loading the Custom Bootloader and Application to Flash Using CCS*

See the loading the custom bootloader and application to Flash using CCS section of the *Live Firmware Update Reference Design with C2000™ Real-Time MCUs* design guide for more information on how to import, build, and load the custom bootloader and application to Flash using CCS.

**Note**

The custom bootloader is located at `<C2000Ware_DigitalPower_SDK_path>\solutions\tida_010062\f28003x\examples\flash\CCS`

Applicable build configurations include Bank0_LDFU and Bank1_LDFU.

Make sure that Bank0_LDFU is built *before* Bank1_LDFU. Once both build configurations are completed, proceed to the follow steps to the flash bootloader and application to the device.

#### *3.1.5.5 Running the LFU Demonstration With Control Loop Running on the CLA and Test Results*

With both flash banks of the device programmed with the custom bootloader and Application images, the LFU demonstration is now ready to run in *Standalone* mode.

To confirm successful switchover, a variety of different signals, registers and LEDs on the device can be observed.

**WARNING**

Confirm the running of new firmware by probing different waveforms such as PWM gate drive signals, input voltage, and current and output voltage using an oscilloscope. Take appropriate safety precautions and appropriate grounding requirements while probing these high voltages and high currents for this isolated DC/DC converter.

Apply power to the board. At this point, verify that LED2 (GPIO34 – D2) is OFF and LED1 (GPIO31 -D1) is ON. This indicates that the Bank0 application is the active application being run. If both LEDs (D1 and D2) are ON, then Bank1 is the active application. Observe that the PWMs gate driver signals are not visible on an oscilloscope. This is due to the software waiting for the SCI autobaud lock command from the Microsoft® Windows® PC.

To enable SCI autobaud lock, execute a command from the Microsoft Windows command prompt. This command is similar to the command previously entered when programming bootloader and application. An example command follows:

- `serial_flash_programmer_appln.exe -d f28003x -k f28003x_fw_upgrade_example\flash_kernel_ex3_sci_flash_kernel_bank0.txt -a f28003x_fw_upgrade_example\llc_f28003x_BANK1FLASH.txt -b 9600 -p COM5`

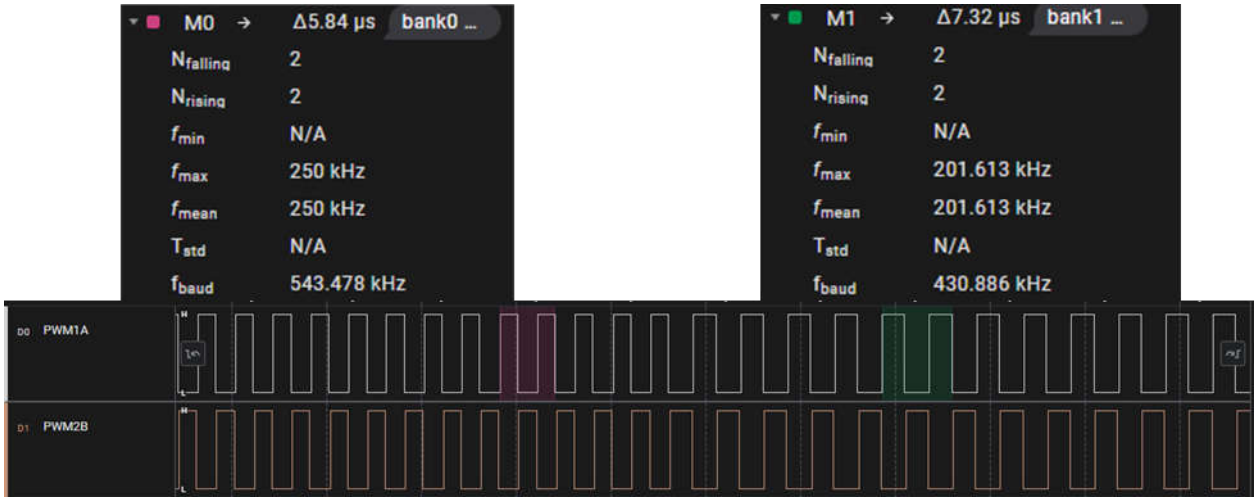



**Figure 3-38. Switching Frequency Change Pre- and Post-LFU Operations**

Now observe on the controlCARD that the LED1 toggles, thus confirming the execution of the LLC_HAL_toggleLED () function in the background task B1 in llc_main.c. LED2 is OFF if Bank0 has the active application and ON if Bank1 has the active application.

The command prompt window shows the same option flash programming options as previously shown. The device is now ready to switch to the other flash bank. To switch execution to the other bank, Enter “8 – Live DFU” – this programs the new firmware to the inactive bank. When complete, Enter “0 – Done” to complete the process. During downloading to Flash, LED1 stops toggling because background tasks are halted.

After programming the new image to the device, LED2 and LED1 switch from ON to OFF (Bank 1 -> Bank 0) or from OFF to ON (Bank0 -> Bank1). This indication of execution of the new image.

The PWM signal waveforms should now be updated with the new targeted switching frequency (200 kHz) as shown in green (M1), see Figure 3-38.

Another step that can be confirmed is the programming of new coefficient for the DF22 controller, this can be done by loading the symbols of Bank0 application into CCS prior to LFU and looking at the contents of the LLC_ctrl_DF22 structure. Specifically observing the targeted control loop parameters b0, b1, and b2. Once the LFU is complete, the symbols for the Bank1 application can be loaded into CCS and the targeted parameters can observed to confirm the switchover and execution of the new image and the appropriate settings. Figure 3-39 illustrates the coefficient values prior and post LFU.

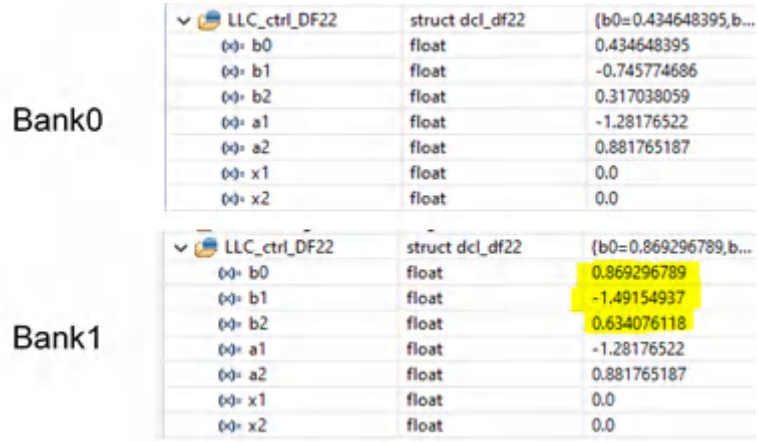



**Figure 3-39. Control Loop Parameter Changes Between Different Images on Bank0 vs Bank1**

**Note**

For the purposes of testing, the coefficients in Bank0 are chosen to be roughly half the values of those shown in Bank1. This is done to demonstrate the impact of a poorly tuned compensator (Bank0) versus a fine-tuned compensator (Bank1). For more robust testing, the designer can perform a transient response analysis and monitor the impact and changes in real-time as the new image is loaded and switched over to.

## 3.2 Testing and Results

The test results are divided into multiple sections that cover the steady state performance, functional performance waveforms and test data, and transient performance waveforms.

### 3.2.1 Performance, Data, and Curve

#### *3.2.1.1 Efficiency, iTHD, and PF of the PFC Stage*

Table 3-5 shows the efficiency data of the PFC stage at a 230-V AC input.

**Table 3-5. Efficiency, iTHD, and PF of the PFC Stage at 230-V AC Input (Without CTRL and DRV Loss)**

| $V_{IN}$ ($V_{ac}$) | $I_{IN}$ ($A_{ac}$) | $P_{IN}$ (W) | $V_{OUT}$ (V) | $I_{OUT}$ (A) | $P_{OUT}$ (W) | EFFICIENCY WITHOUT CTRL AND DRV | EFFICIENCY WITH CTRL AND DRV | iTHD (%) | PF |
|---|---|---|---|---|---|---|---|---|---|
| 230.00 | 0.56 | 108 | 386.38 | 0.273 | 105.48 | 97.65% | 95.80% | 11.50% | 0.997 |
| 229.93 | 1.301 | 238 | 386.39 | 0.607 | 234.54 | 98.41% | 97.56% | 4.76% | 0.999 |
| 229.87 | 1.691 | 360 | 386.39 | 0.919 | 355.09 | 98.60% | 98.03% | 3.55% | 0.999 |
| 229.8 | 2.321 | 481 | 386.33 | 1.228 | 474.41 | 98.73% | 98.31% | 2.41% | 0.999 |
| 229.74 | 2.799 | 602 | 386.41 | 1.540 | 595.07 | 98.82% | 98.47% | 2.19% | 0.999 |
| 229.68 | 3.258 | 723 | 386.52 | 1.849 | 714.68 | 98.82% | 98.54% | 1.82% | 0.999 |
| 229.61 | 3.706 | 846 | 386.39 | 2.165 | 836.53 | 98.83% | 98.59% | 1.41% | 0.999 |
| 229.55 | 4.237 | 969 | 386.52 | 2.475 | 956.64 | 98.78% | 98.56% | 1.25% | 0.999 |
| 229.53 | 4.414 | 1006 | 386.31 | 2.569 | 992.43 | 98.65% | 98.45% | 1.38% | 0.999 |

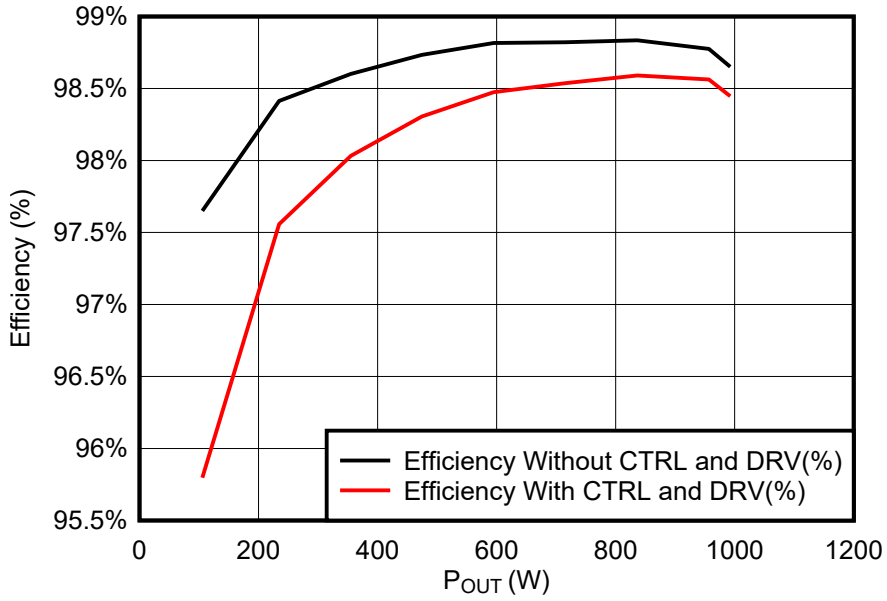



**Figure 3-40. Efficiency of the PFC Stage at 230-V AC**

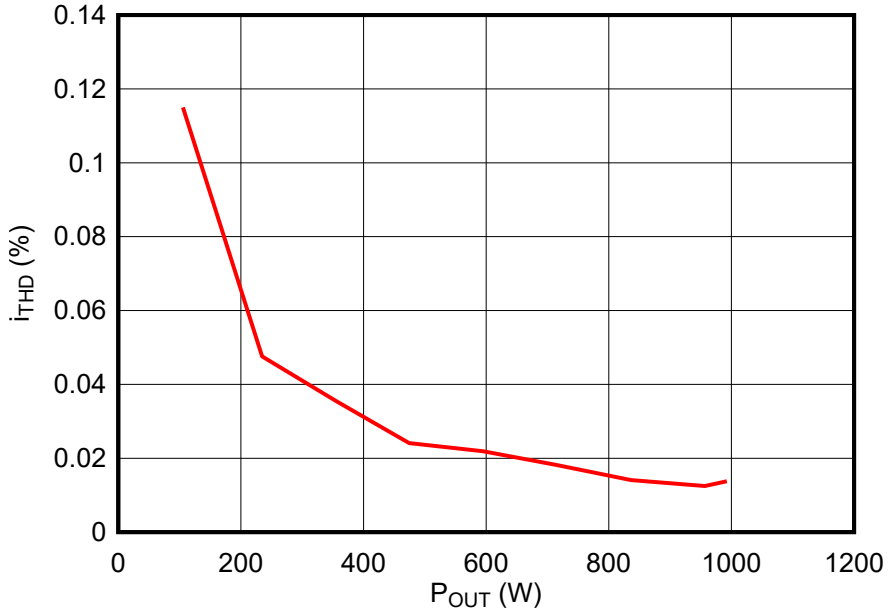



**Figure 3-41. iTHD at 230-V AC**

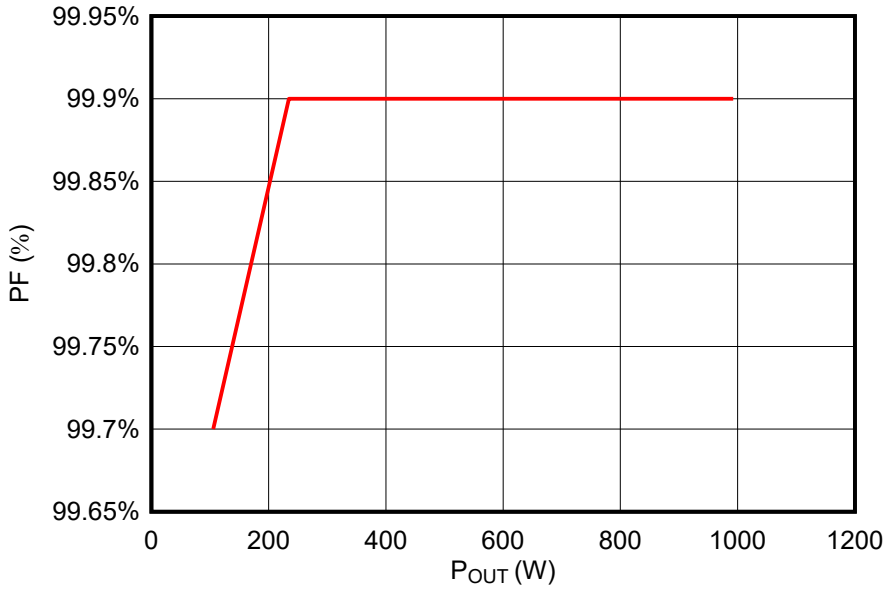



**Figure 3-42. PF at 230-V AC**

### *3.2.1.2 Efficiency of the LLC Stage*

Table 3-6 shows the efficiency of the LLC stage with an open-loop test.

**Table 3-6. Efficiency of the LLC stage**

| $V_{IN}$ (V) | $I_{IN}$ (A) | $V_O$ (V) | $I_O$ (A) | $P_O$ | EFFICIENCY WITHOUT CTRL AND DRV | EFFICIENCY WITH CTRL AND DRV |
|---|---|---|---|---|---|---|
| 385.9 | 0.324 | 12.08 | 10 | 121 | 96.61% | 95.16% |
| 385.9 | 0.636 | 12.05 | 20 | 241 | 98.20% | 97.45% |
| 385.8 | 0.949 | 12.02 | 30 | 361 | 98.49% | 97.98% |
| 385.8 | 1.262 | 12.00 | 40 | 480 | 98.59% | 98.21% |
| 385.8 | 1.575 | 11.96 | 50 | 598 | 98.43% | 98.12% |
| 385.7 | 1.888 | 11.95 | 60 | 717 | 98.46% | 98.20% |
| 385.7 | 2.200 | 11.91 | 70 | 834 | 98.25% | 98.03% |

**Table 3-6. Efficiency of the LLC stage (continued)**

| $V_{IN}$ (V) | $I_{IN}$ (A) | $V_O$ (V) | $I_O$ (A) | $P_O$ | EFFICIENCY WITHOUT CTRL AND DRV | EFFICIENCY WITH CTRL AND DRV |
|---|---|---|---|---|---|---|
| 385.7 | 2.513 | 11.89 | 80 | 951 | 98.15% | 97.95% |
| 385.7 | 2.670 | 11.87 | 85 | 1009 | 97.99% | 97.81% |

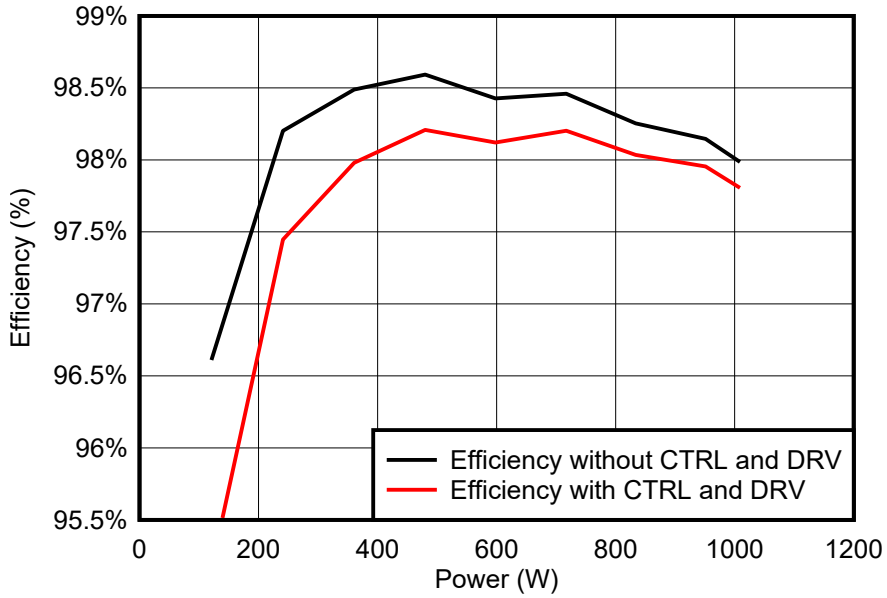



**Figure 3-43. Efficiency of the LLC Stage**

### *3.2.1.3 Efficiency of the Whole System*

Table 3-7 shows the system efficiency of two stages without control loss.

**Table 3-7. Efficiency of the Whole System**

| $V_{IN}$ (V) | $I_{IN}$ (A) | $P_{IN}$ (W) | THD (%) | PF | $V_O$ (V) | $I_O$ (A) | $V_{BUS}$ (V) | EFFICIENCY WITHOUT DRV AND CTRL (%) | $P_{DRV}$ (W) | EFFICIENCY WITH ONLY $P_{DRV}$ (%) |
|---|---|---|---|---|---|---|---|---|---|---|
| 230.08 | 0.623 | 126.97 | 9.23% | 0.885 | 12.12 | 10 | 386 | 95.46% | 3.106 | 93.18% |
| 239.85 | 1.123 | 249.38 | 5.35% | 0.965 | 12.09 | 20 | 386 | 96.96% | 3.106 | 95.77% |
| 229.89 | 1.657 | 371.67 | 3.51% | 0.975 | 12.07 | 30 | 386 | 97.43% | 3.106 | 96.62% |
| 229.82 | 2.328 | 493.9 | 2.60% | 0.923 | 12.03 | 40 | 386 | 97.43% | 3.106 | 96.82% |
| 229.76 | 2.829 | 616.4 | 2.07% | 0.947 | 12.00 | 50 | 386 | 97.34% | 3.106 | 96.85% |
| 229.70 | 3.286 | 738.5 | 1.46% | 0.978 | 11.97 | 60 | 386 | 97.25% | 3.106 | 96.84% |
| 229.63 | 3.805 | 861.4 | 1.46% | 0.985 | 11.95 | 70 | 386 | 97.11% | 3.106 | 96.76% |
| 229.57 | 4.302 | 984.1 | 1.07% | 0.996 | 11.95 | 80 | 386 | 96.90% | 3.106 | 96.60% |
| 229.54 | 4.576 | 1046 | 1.23% | 0.996 | 11.92 | 85 | 386 | 96.86% | 3.106 | 96.58% |

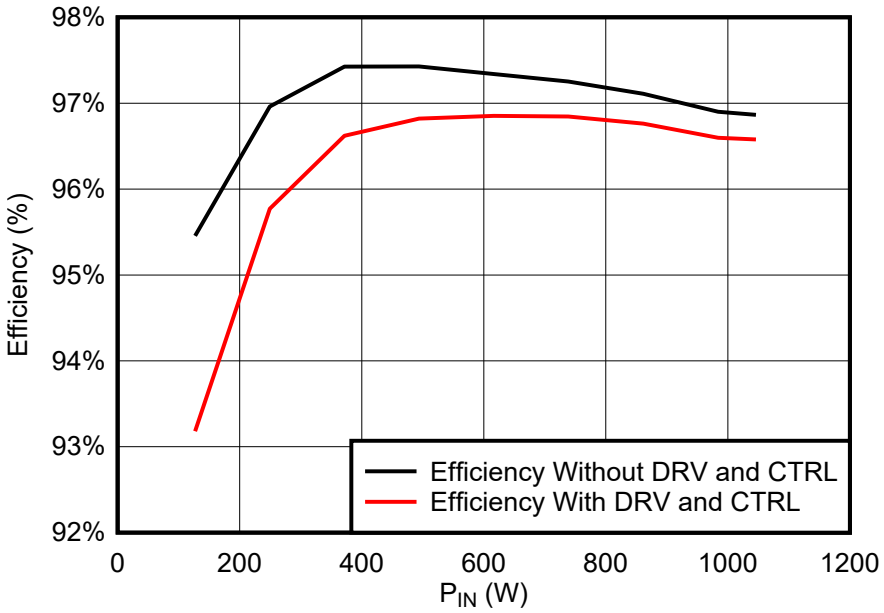



**Figure 3-44. Efficiency of the Whole System**

### 3.2.2 Functional Waveforms

#### *3.2.2.1 Start-up*

Figure 3-45 shows the no-load, start-up waveform. Figure 3-46 shows the half-load, start-up waveform.

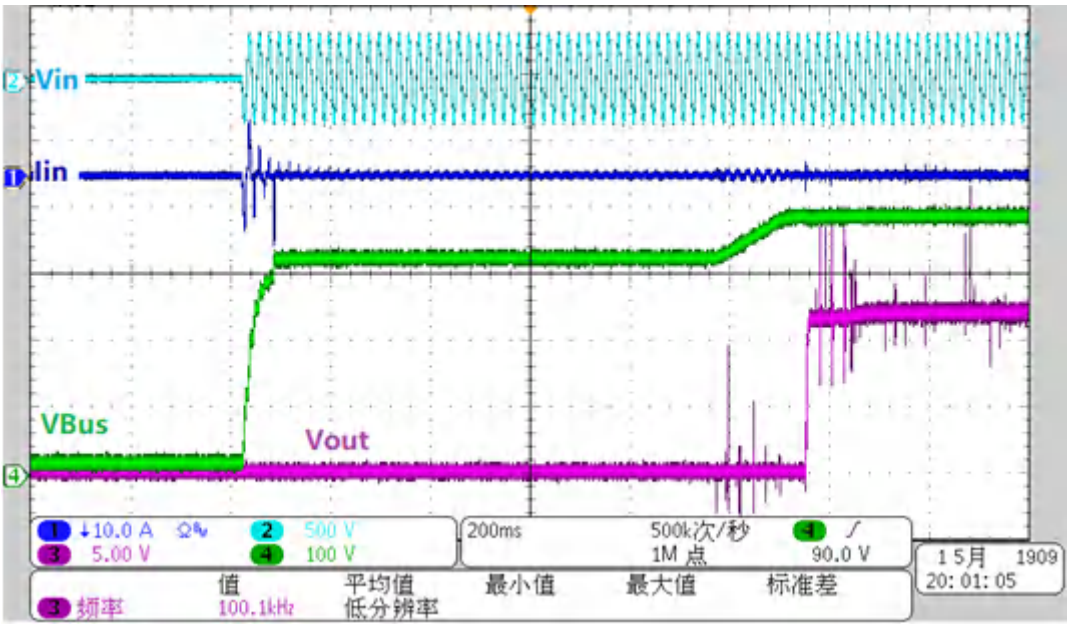



**Figure 3-45. System Start-up With No Load**

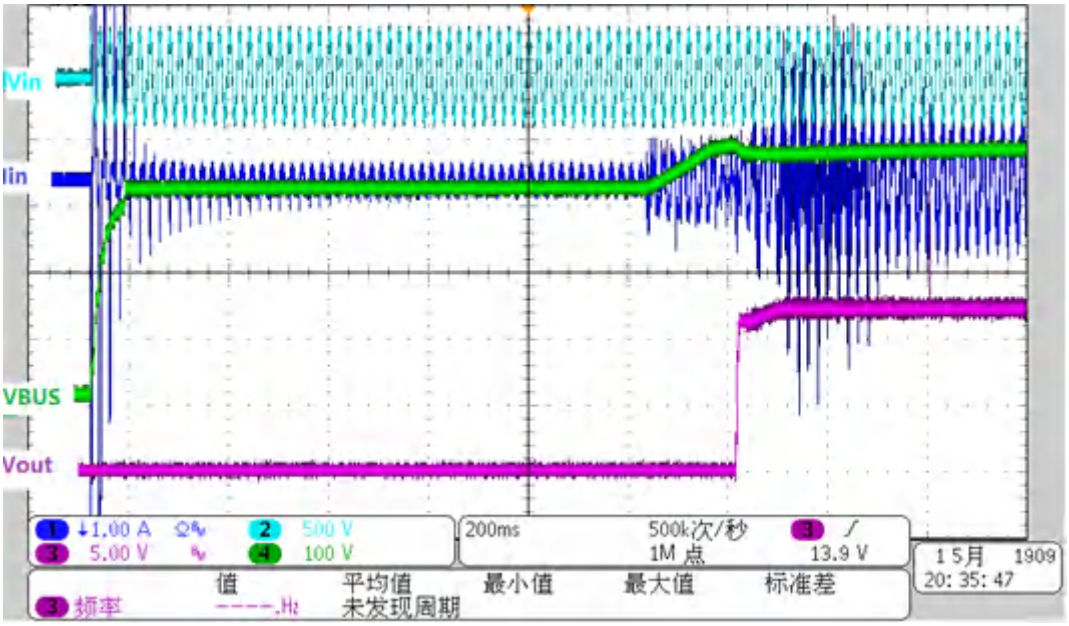



**Figure 3-46. System Start-up With 50% Load**

### 3.2.2.2 Hall Sensor

Figure 3-47 shows the output of the TMCS1100 Hall sensor corresponding with the AC input current.

Figure 3-48 shows the ADC sample value corresponding with the actual current flow through the primary Hall sensor, which proves the Hall sensor linearity is precise.

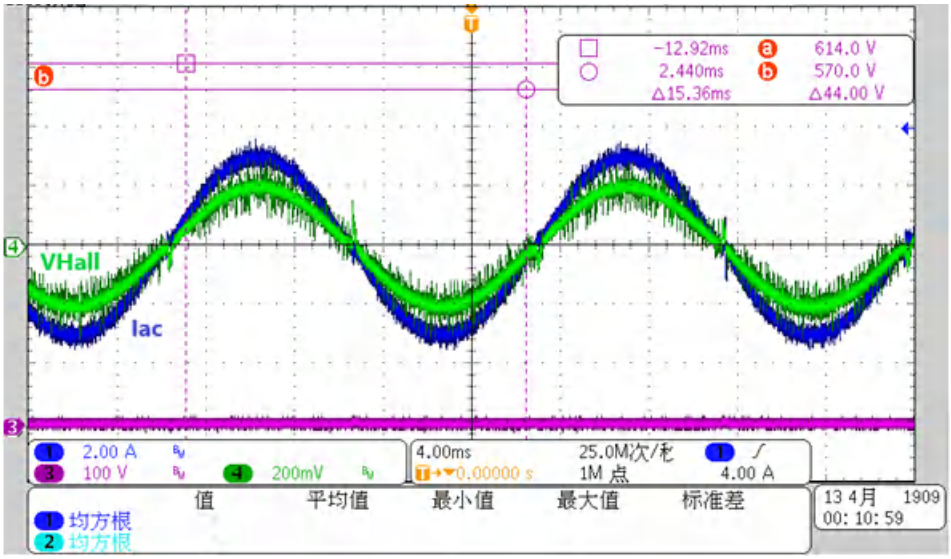



**Figure 3-47. TMCS1100 Hall-Sensing Waveform**

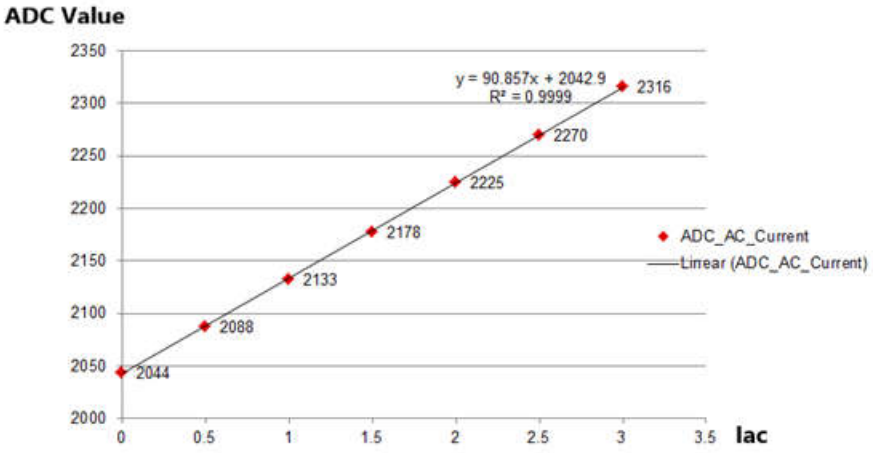



**Figure 3-48. ADC Value Corresponding With Actual Current**

### 3.2.2.3 PFC Working Waveforms

This section shows the PFC input and switching waveforms. Figure 3-49 and Figure 3-50 show the input voltage and current at 230-V AC input with no load and 20% load, and Figure 3-51 shows the switching waveform of the PFC stage with a GaN card.

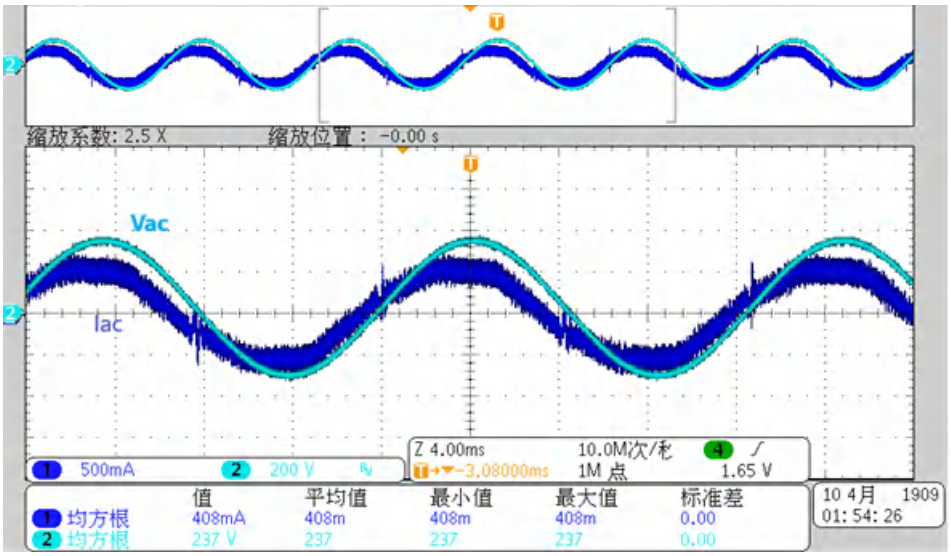



**Figure 3-49. 230-V AC, No Load Input Voltage and Current**

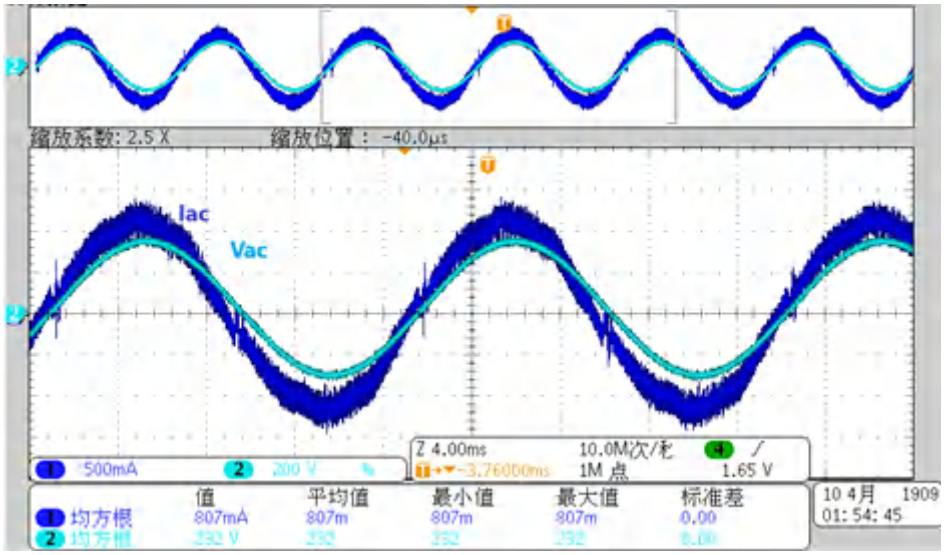



**Figure 3-50. 230-V AC, 20% Load Input Voltage and Current**

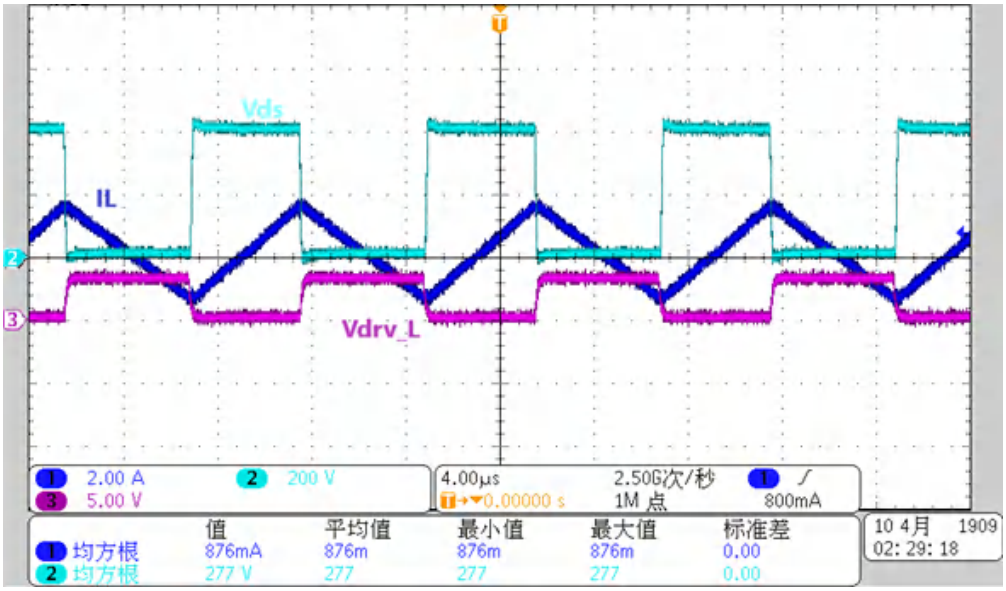



**Figure 3-51. PFC Switching Waveform**

### *3.2.2.4 LLC Working Waveforms*

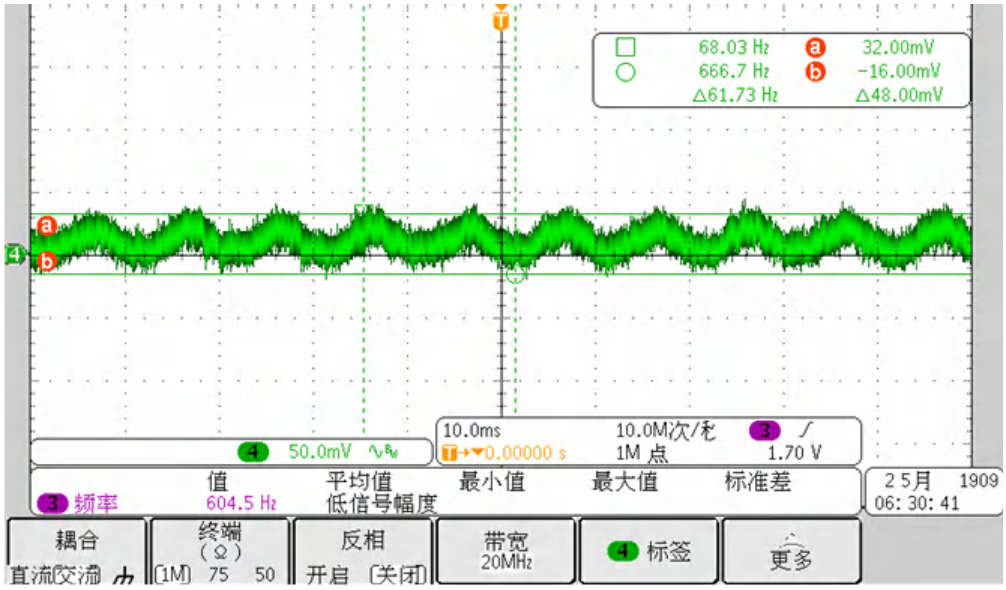



**Figure 3-52. Current Ripple 50% Load**

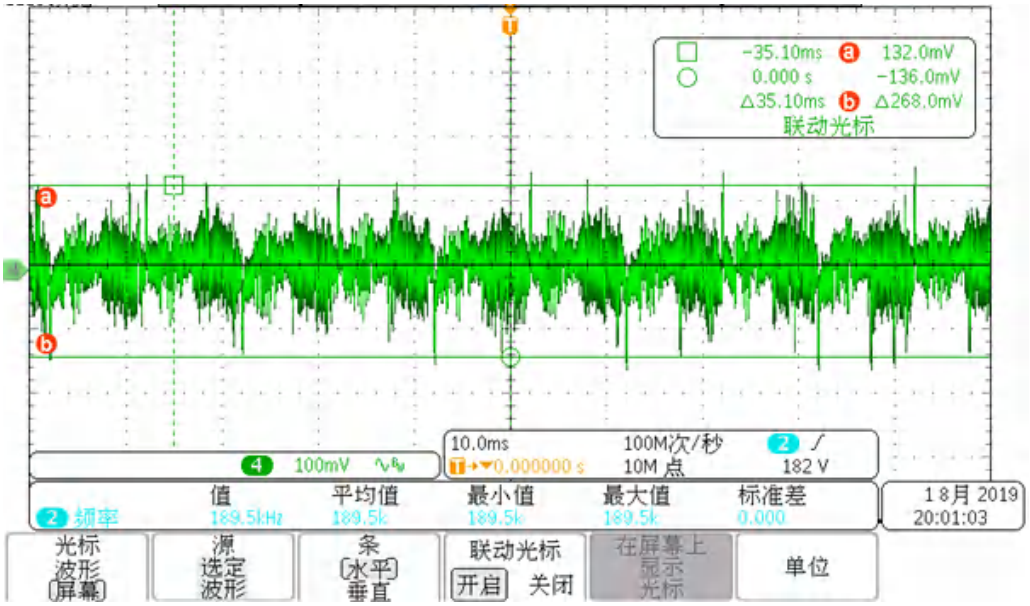



**Figure 3-53. Current Ripple 100% Load**

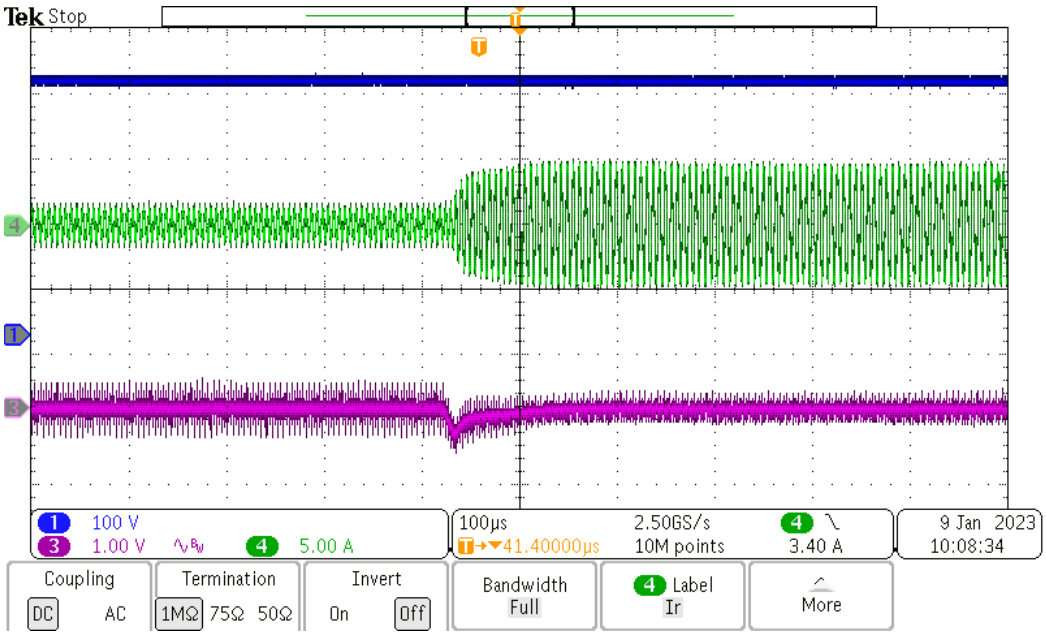



**Figure 3-54. Load Transient 0%–50%**

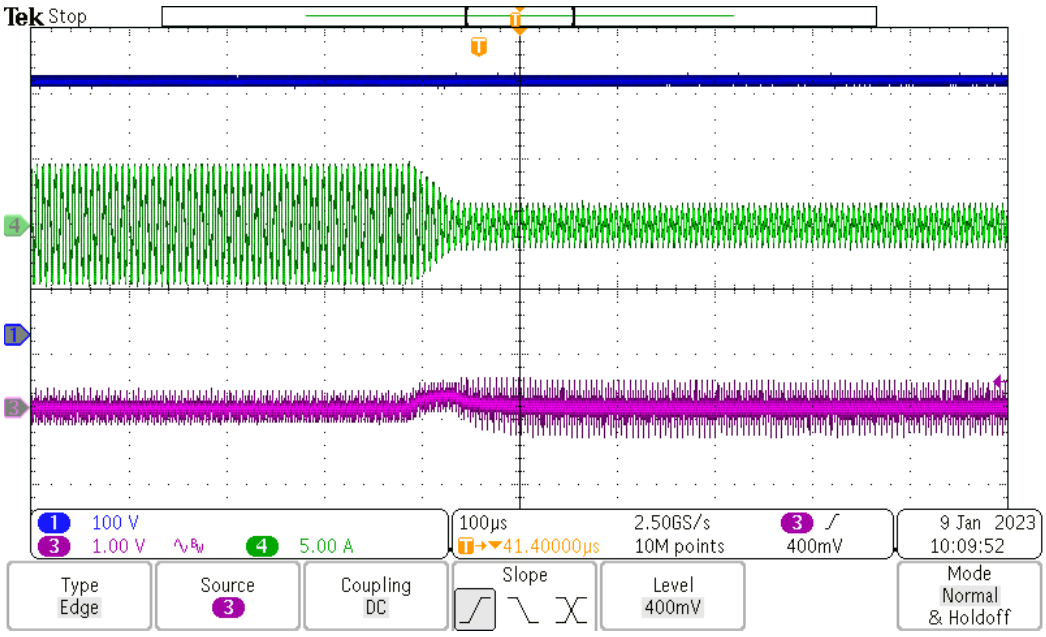



**Figure 3-55. Load Transient 50%–0%**

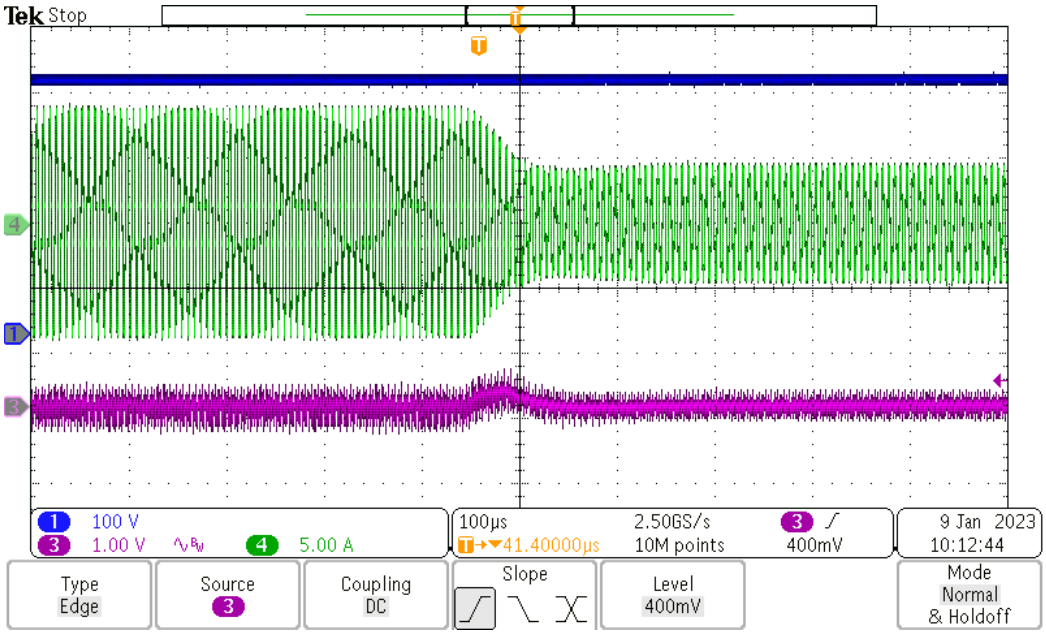



**Figure 3-56. Load Transient 100%–50%**

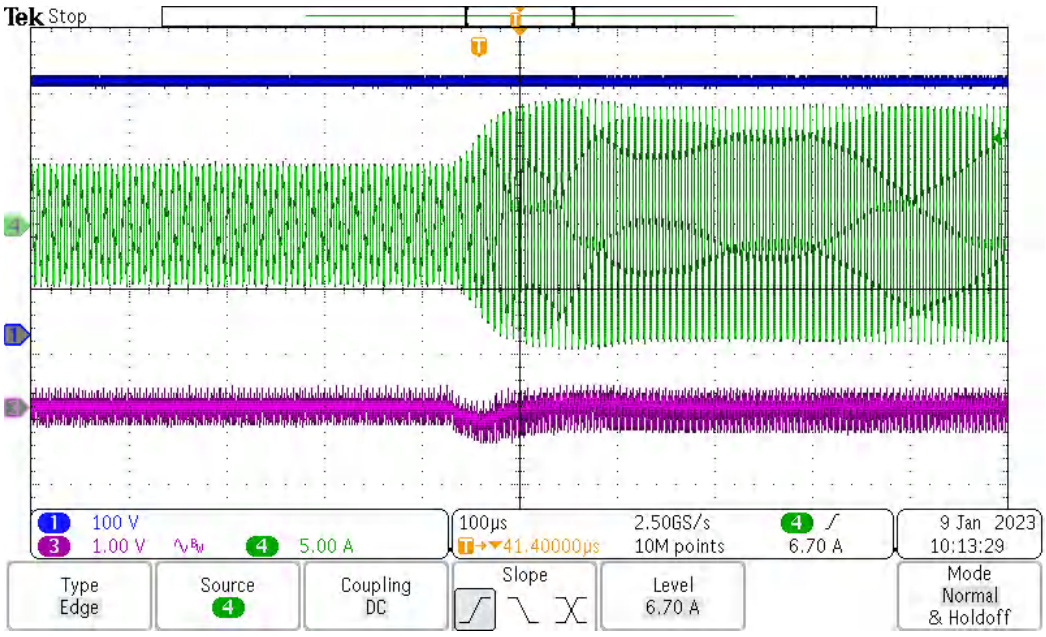



**Figure 3-57. Load Transient 50%–100%**

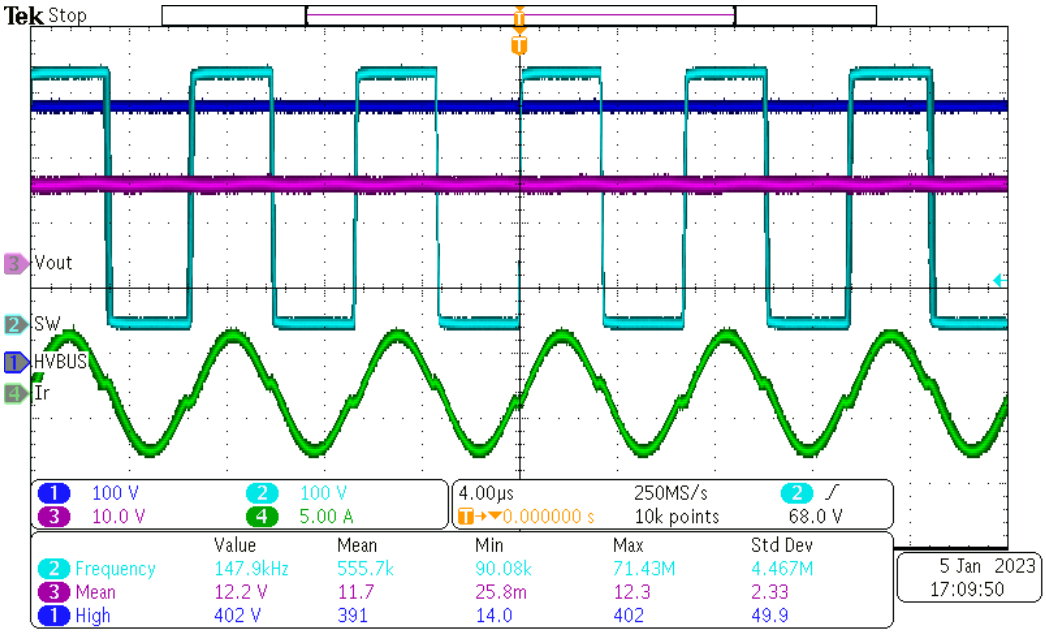



**Figure 3-58. LLC Stage Switching Waveform**

# 4 Design Files

## 4.1 Schematics

To download the schematics, see the design files at TIDA-010062.

## 4.2 Bill of Materials

To download the bill of materials (BOM), see the design files at TIDA-010062.

## 4.3 PCB Layout Recommendations

A careful PCB layout is critical in a high-current, fast-switching circuit to provide appropriate device operation and design robustness. As with all switching power supplies, pay attention to detail in the layout to save time in troubleshooting.

### 4.3.1 Power Stage Specific Guidelines

Follow these key guidelines to route the power stage components:

- Minimize the loop area and trace length of the power path circuits, which contain high-frequency switching currents. This will help reduce EMI and improve the converter's overall performance.
- Keep the switch node as short as possible. A short and optimal trace width helps reduce induced ringing caused by parasitic inductance.
- Keep traces with high dV/dt potential and high di/dt capability away from or shielded from sensitive signal traces with adequate clearance and ground shielding.
- Keep the power ground and control ground separate for each power supply stage. Tie them together (if they are electrically connected) in one point near the DC input return or output return of the given stage.
- When multiple capacitors are used in parallel for current sharing, the layout should be symmetrical across both capacitor leads. If the layout is not identical, the capacitor with the lower series trace impedance will see higher peak currents, and become hotter (I2R).
- Tie the heat sinks of all of the power switching components to their respective power grounds.
- Place protection devices such as TVS, snubbers, capacitors, or diodes physically close to the device that they are intended to protect, and route with short traces to reduce inductance.
- Choose the width of PCB traces based on an acceptable temperature rise at the rated current as per IPC2152, as well as acceptable DC and AC impedances. The traces should withstand the fault currents (such as short circuit current) before electronic protection devices, such as fuses or circuit breakers, are activated.
- Determine the distances between various traces of the circuit, according to the requirements of applicable standards. For this design, the UL 60950-1 safety standard is followed to maintain the creepage and clearance from live line, to neutral line, and to safety ground.
- Adapt the thermal management to fit the end equipment.

### 4.3.2 Gate Driver Specific Guidelines

Follow these key guidelines to route the high-frequency, high-current gate driver:

- Place the driver device as close as possible to the power device to minimize the length of high current traces between the output pins of gate drive and the gate of the power device.
- Locate the VDD bypass capacitors between VDD and GND as close as possible to the driver with minimal trace length to improve the noise filtering. These capacitors support high-peak current being drawn from VDD.
- Minimize the turn-on and turn-off current-loop paths (driver device, power MOSFET, and VDD bypass capacitor) as much as possible to keep the stray inductance to a minimum.
- Minimize noise coupling with star point grounding from one current loop to another. Connect the driver GND to the other circuit nodes, such as the power switch source or the PWM controller ground, at one single point. The connected paths must be as short as possible to reduce inductance and be as wide as possible to reduce resistance.

### 4.3.3 Layout Prints

To download the layer plots, see the design files at TIDA-010062.

### 4.4 Altium Project

To download the Altium Designer® project files, see the design files at TIDA-010062.

### 4.5 Gerber Files

To download the Gerber files, see the design files at TIDA-010062.

### 4.6 Assembly Drawings

To download the assembly drawings, see the design files at TIDA-010062.

## 5 Software Files

To download the software files, see the design files at TIDA-010062.

## 6 Related Documentation

1. Texas Instruments, *Interleaved CCM Totem Pole Bridgeless PFC Reference Design Using C2000™ MCU*
2. Texas Instruments, *Designing an LLC Resonant Half-Bridge Power Converter Seminar*
3. Texas Instruments, *Resonant LLC Half-Bridge DC/DC Converter Software Design Guide*
4. Texas Instruments, *Resonant LLC Half-Bridge DC/DC Converter Hardware Design Guide*
5. Texas Instruments, *LMG341xR070 600-V 70-mΩ GaN with Integrated Driver and Protection Data Sheet*
6. Texas Instruments, *TMS320F28004x Piccolo™ Microcontrollers Data Manual*
7. Texas Instruments, *TMS320F28004x Real-Time Microcontrollers Technical Reference Manual*
8. Texas Instruments, *TMS320F28003x Real-Time Microcontrollers Technical Reference Manual*
9. Texas Instruments, *TMS320F28002x Real-Time Microcontrollers Technical Reference Manual*

### 6.1 Trademarks



## 7 About the Author

**DESHENG GUO** is a System Application Engineer at Texas Instruments, where he is responsible for developing customized power solutions as part of the power delivery industrial segment. Desheng brings to this role his extensive experience in power electronics, power conversion, EMI and EMC, power and signal integrity, and analog circuits design spanning many high-profile organizations. He received his master's degree from the Harbin Institute of Technology with Power electronics in 2007, and has been working in DPEC of DELTA for many years, focusing on research and design of high-efficiency power supply.

**MINGHAN DONG** is a Systems Engineer at Texas Instruments where he is responsible for developing digital reference design solutions for the power design service, industrial segment. Minghan focuses on C2000 firmware design and analog power circuit design. Minghan earned his master's degree of electrical and electronics engineering from Zhejiang University, Hangzhou.

**JINHAN ZENG** is a Analog Field Application Engineer at Texas Instruments where he is responsible for industrial customers and provides them with the most suitable system solutions. He applied GaN to the LLC part of this design, improved the system performance of the entire system by 0.3%, and provided customers with GaN power data. Jinhan is focused on analog sockets and switching power applications and has experience working in Huawei. Jinhan got his bachelor's degree of electronics engineering from Shenzhen University, Shenzhen.

# 8 Revision History

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

| Changes from Revision F (March 2023) to Revision G (October 2023) | Page |
|---|---|
| • Updated the document title | 1 |
| • Added *Live Firmware Update* in Section 3.1.5 | 47 |

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

| Changes from Revision E (March 2022) to Revision F (March 2023) | Page |
|---|---|
| • Replaced Si MOSFETs with LMG342x GaN in LLC part | 1 |

| Changes from Revision D (January 2022) to Revision E (March 2022) | Page |
|---|---|
| • Added TMS320F280039 product folder to the *Resources* section | 1 |
| • Added TMS320F280039 | 3 |
| • Added F28003x | 15 |
| • Added F28003x | 19 |
| • Added llc_F28003x | 33 |

| Changes from Revision C (March 2021) to Revision D (January 2022) | Page |
|---|---|
| • Added TMDSFSIADAPEVM product folder to the *Resources* section | 1 |
| • Added FSI feature in the *Features* section | 1 |
| • Added F280039 to the *Communication Between the Primary Side and the Secondary Side* section | 11 |
| • Removed the *System Test: Dual Stages* section from Section 3.1.1.3 | 13 |
| • Added *PFC + LLC Stage Dual Test* section | 44 |

| Changes from Revision B (September 2019) to Revision C (March 2021) | Page |
|---|---|
| • Updated the numbering format for tables, figures, and cross-references throughout the document. | 1 |
| • Required Hardware and Software section completely reworked | 12 |

| Changes from Revision A (September 2019) to Revision B (September 2019) | Page |
|---|---|
| • Changed title | 1 |
| • Changed description | 1 |
| • Changed features | 1 |

| Changes from Revision * (September 2019) to Revision A (September 2019) | Page |
|---|---|
| • Changed description | 1 |

# 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