EVM User's Guide: INA4230EVM INA4235EVM
INA423x Evaluation Module

Description

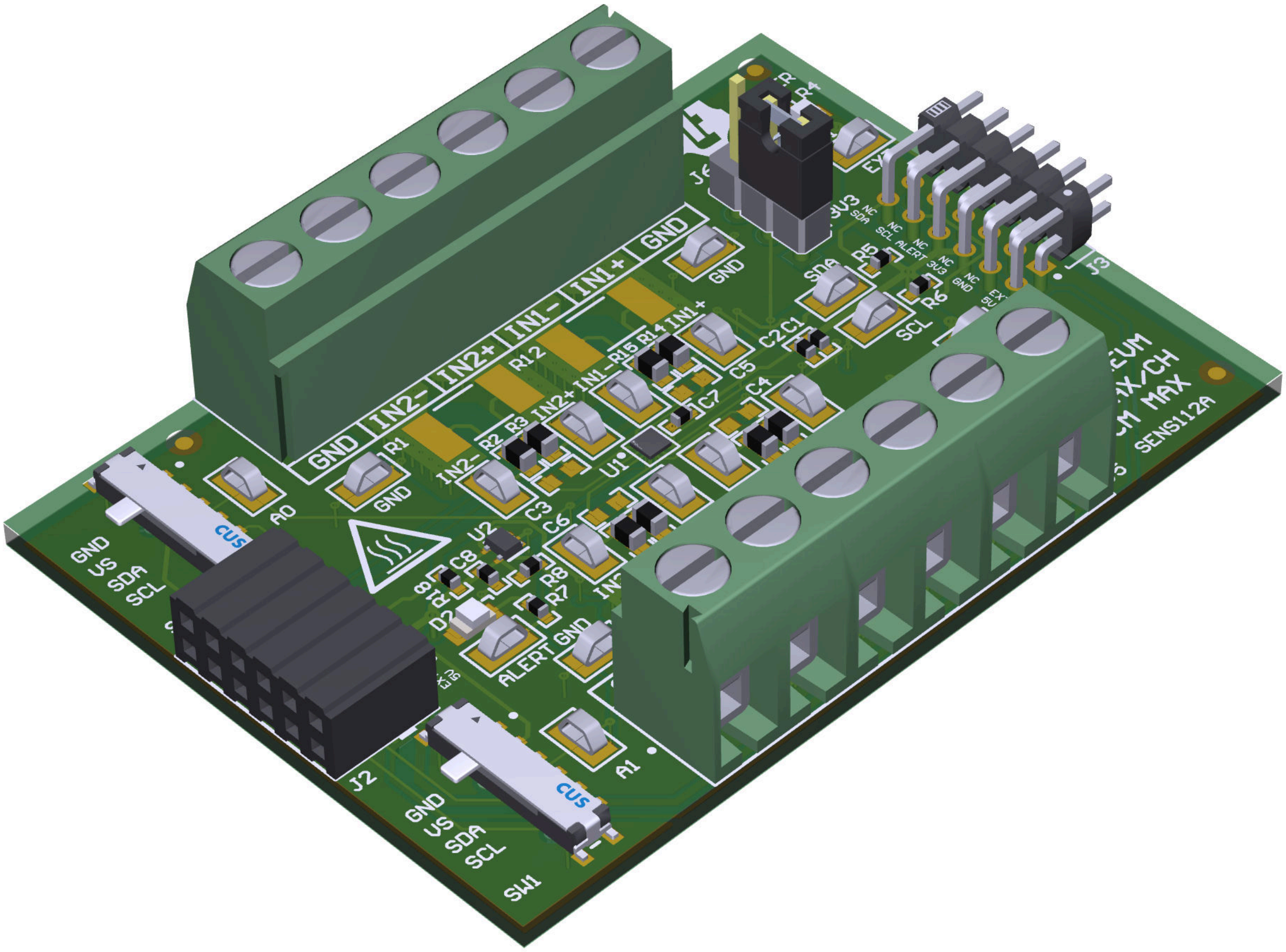
The INA423X evaluation module (EVM) is a platform to evaluate the main features and performance of the INA4230 and INA4235. The EVM supports current measurements up to 10A per channel and comes with graphical user interface (GUI) support to read and write to device registers.

Get Started

1. Buy the INA4230EVM or INA4235EVM evaluation board.
2. Buy the TI-SCB sensor control board.
3. Download and install the PAMB Windows USB Drivers.
4. Read this user's guide to set up the hardware.
5. Run the INA423XEVM-GUI.

Features

- GUI support to read and write device registers as well as view and save results data
- EVM detached from SCB for custom use cases
- Multiple EVM support with single SCB/GUI
- Conveniently powered from a common micro-USB connector through the SCB



1 Evaluation Module Overview

1.1 Introduction

The EVM is an easy-to-use platform for evaluating the main features and performance of the INA4230 or INA4235. The EVM supports current measurements up to 10 amps per channel through the PCB, and includes a graphical user interface (GUI) used to read and write device registers as well as view and save results data.

This user's guide describes the characteristics, operation, and use of the INA4230 and INA4235 evaluation modules (EVMs). These EVMs are designed to evaluate the performance of the INA4230 and INA4235. Throughout this document, the terms evaluation board, evaluation module, and EVM are synonymous with the INA4230EVM and INA4235EVM. This document includes a schematic, reference printed-circuit board (PCB) layouts, and a complete bill of materials (BOM).

1.2 Kit Contents

Table 1-1 lists the contents of the EVM kit. Contact Texas Instruments Customer Support if any component is missing.

Table 1-1. Kit Contents

ITEM	QUANTITY
INA4230EVM or INA4235EVM	1

Note that this EVM requires the TI Sensor Control Board (SCB), which is sold separately and can be found here.

1.3 Specification

The EVM is intended to provide basic functional evaluation of the devices. The layout is not intended to be a model for the target circuit, nor laid out for electromagnetic compatibility (EMC) testing. The EVM consists of one printed-circuit board (PCB). The PCB is referred to as the EVM, and has either the INA4230 or INA4235 installed.

1.4 Device Information

The INA4235 and INA4230 device is are quad channel 16-bit digital current monitors with an I2C/SMBus-compatible interface that are compliant with digital bus voltages from 1.2V to 5V. The devices monitor the voltage across an external sense resistor and reports values for shunt voltage, bus voltage, current, power, and energy for each channel.

The INA4235 and INA4230 feature programmable ADC conversion times and averaging that is common for all channels. Each channel has a programmable calibration value with an internal multiplier that enables direct readouts of current in amperes, power in watts, and energy in joules. Each channel monitors the bus voltage present on the IN– pin and can alert on overcurrent and undercurrent conditions, as well as on overvoltage and undervoltage conditions. High input impedance while in current measurement mode allows use of larger current sense resistors needed to measure small value system currents.

The INA4235 and INA4230 sense current on common-mode bus voltages that can vary from –0.3V to 48V, independent of the supply voltage. The devices operate from a single 1.7V to 5.5V supply, drawing a typical supply current of 400µA in normal operation. The devices can be placed in a low-power standby mode where the typical operating current is 2.5µA and can be fully disabled using the enable pin to achieve a supply current less than 50nA (INA4235) or 1µA (INA4230). The devices are specified over the operating temperature range between –40°C and 125°C and features up to 16 programmable addresses.

Table 1-2. Device Summary

PRODUCT	DIGITAL PROTOCOL	ADC RESOLUTION	MAX GAIN ERROR	MAX OFFSET VOLTAGE
INA4230	I2C	16-bit	0.75%	±75 µV
INA4235	I2C	16-bit	0.1%	±10 µV

2 Hardware

2.1 Quick Start Setup

The following instructions describe how to set up and use the EVM.

1. Purchase an SCB if you do not already have one.
   a. To use a PAMB Controller instead, see *PAMB Compatibility*.
2. Download this driver and install as an administrator: https://www.ti.com/lit/zip/sbac253.
   a. Follow the download prompts; a myTI account is required.
   b. Note that this driver is labeled as a PAMB driver, but is also used for the SCB.
3. Attach the EVM to the SCB Controller as shown in Figure 2-1.
   a. Refer to Figure 2-2 when connecting multiple EVMs of the same type together.
4. Connect the EVM to the PC using the provided USB cable.
   a. Insert the micro USB cable into the SCB Controller onboard USB receptacle J2.
   b. Plug the other end of the USB cable into a PC.
5. Access the GUI from this link in either Chrome®, Firefox®, or Safari®: https://dev.ti.com/gallery/info/CurrentSensing/INA423XEVM_GUI.
6. Connect the GND reference of the external system to the GND node of the EVM (pins 1 or 6 of J1 or J5).
7. Provide a differential input voltage signal to the IN+ and IN– nodes of the desired channel on either J1 or J5 as explained in *Current Sensing Operation*.

2.2 EVM Operation

To use the EVM with the SCB Controller (sold separately), connect the EVM as shown in Figure 2-1.

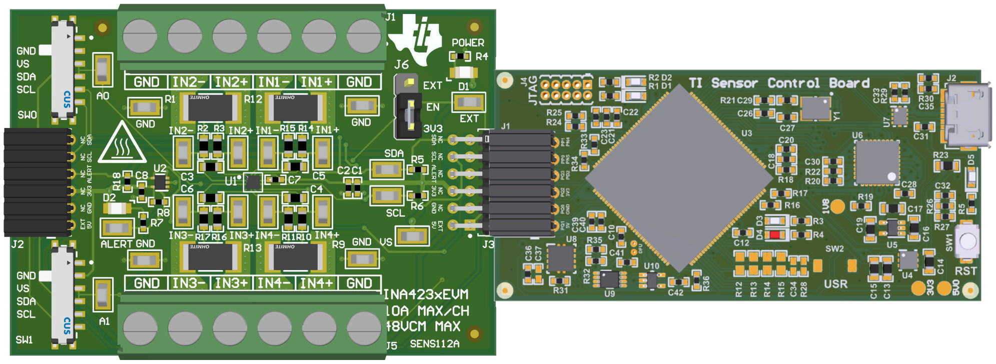


Figure 2-1. EVM (Left) Connected to SCB Controller (Right)

If using multiple EVMs, then connect as shown in Figure 2-2. Make sure to use a different address for each device. The GUI only supports one EVM and device type at a time; up to four EVMs total.

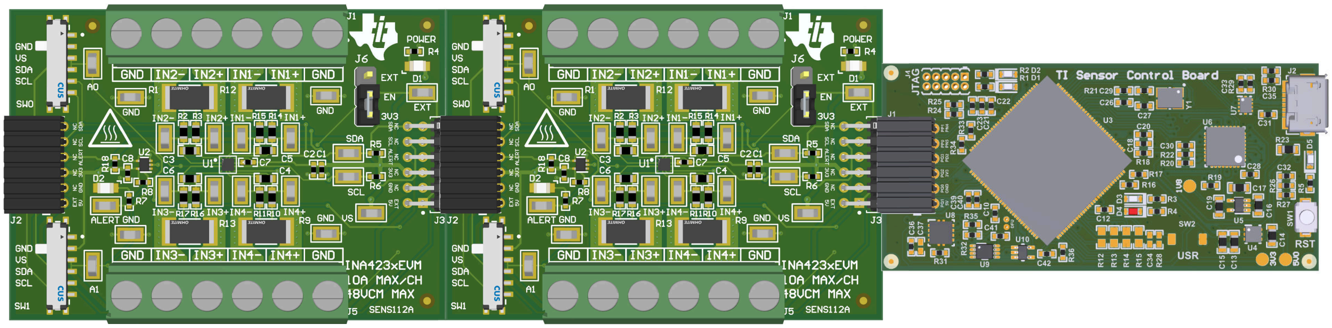

GND | IN2- | IN2+ | IN1- | IN1+ | GND
GND | IN3- | IN3+ | IN4- | IN4+ | GND
INA423xEVM
10A MAX/CH
48VCM MAX
J5 SENS112A
TI Sensor Control Board
USR
RST

Figure 2-2. Multiple EVMs Connected to SCB Controller

2.2.1 Current Sensing Operation

The EVM can be used with either onboard or external shunt resistors. To use an onboard shunt resister, solder a 2512 surface-mount technology (SMT) shunt resistor across the pads of R1, R12, R13 or R9 and connect in series with the external system and load current through J1 or J5. An external shunt can be connected directly across the terminals of J1 or J5.

2.2.1.1 Detailed Setup

To configure a measurement evaluation, follow these steps:

1. Connect a shunt resister by doing either of the following:
   a. Solder a 2512 resistor across the pads of R1, R12, R13, or R9 that connects the IN+ and IN– inputs as shown in Figure 2-4.
   b. Connect an external shunt across the IN+ and IN– terminals of J1 or J5 as shown in Figure 2-3.
      i. If an external shunt is being used, then make the connections such that the sensing location is across the shunt and there is no high current on the sensing path. See the TI Precision Labs - Current Sense Amplifiers: Shunt Resistor Layout video for more information.

2. Connect the IN+ and IN– terminals in series with the load while powered off.
   a. When measuring more than 10A, verify that the high current path does not go through the EVM (including the terminal blocks J1 or J5), as shown in Figure 2-3.

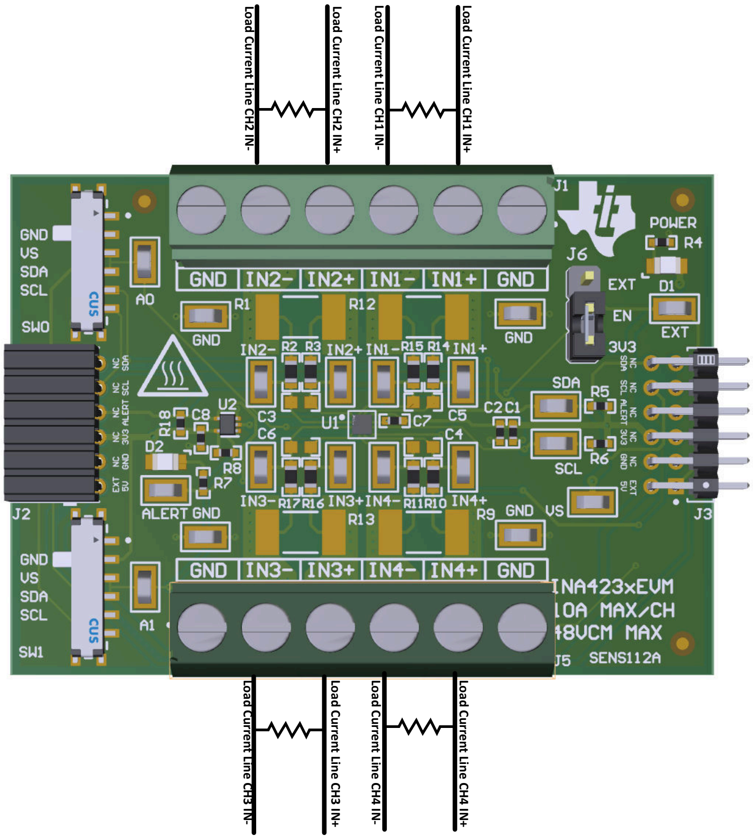


Figure 2-3. IN+ and IN– Wiring for More Than 10A

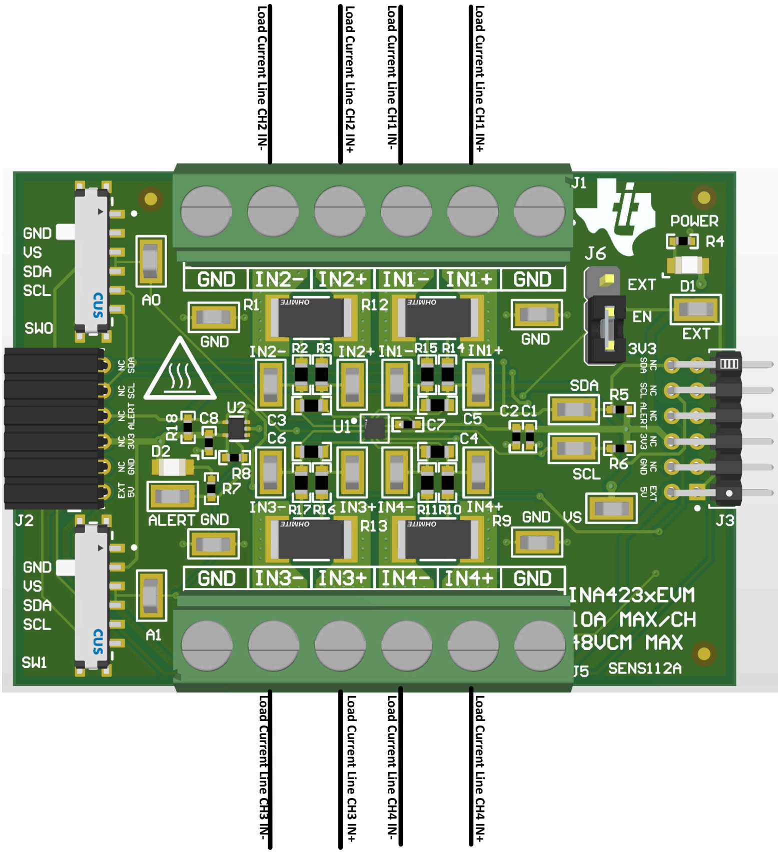


Figure 2-4. IN+ and IN– Wiring for 10A or Less

WARNING

When measuring current, first make sure that the equipment (shunt resistor, wires, connectors, and so on) can support the amperage and power dissipation. Secondly, make sure that the current flowing through J1 and J5 does not exceed 10A per channel. Failure to do so can result in damage to the EVM, or personal injury.

The EVM can get hot.

3. Connect the system ground to either of the GNDs on terminals J1 and J5.
4. Power on the system and observe the device states and outputs through the GUI.

2.3 Circuitry

This section summarizes the EVM subsystems and components.

2.3.1 Current Sensing IC

This section describes the main INA device and supporting components.

U1 is the main INA current-sensing device (either the INA4230 or INA4235). C1, C2, and C7 are bypass capacitors that are placed near the sensor to help mitigate power supply noise and provide current quickly to the device when needed. LED D1 with current limiting resistor R4 are used to indicate when the EVM is powered on.

J6 is used to connect the enable pin to either 3V3 or an external source. By default, jumper SH-J1 is populated across pins 1 and 2 to connect the enable pin to 3V3. Move jumper SH-J1 across pins 2 and 3 to connect to an external source, accessed through test point EXT.

The device pins can be monitored directly through the test points TP1 – TP19. Note that there are two extra test points on GND for convenience.

2.3.2 Input Signal Path

This section describes the circuitry of the input signal path.

J1 and J5 are the main connection terminals. Pins 1 and 6 of J1 and J5 are used to tie the system ground to the EVM ground. Pins 2-5 of J1 and J5 are used for the each channel's IN+ and IN- connection.

R12, R1, R13 or R9 can be used for an optional onboard shunt resistor with a 2512 footprint. Alternatively, a shunt can be placed across the IN+ and IN– terminals of the desired channel on J1 or J5. If desired, a differential voltage can be applied directly for measurement tests.

Each channel has two resistors and a capacitor that combine to make an optional input filter. The resistors are populated with 0Ω resistors by default. When using input filtering, take into account the input bias current of the device. The capacitor can also be used without the resistors to reduce noise. See the **data sheet** for more info on input filtering. Each filter of the channels is as follows:

- Channel 1 - R14, R15, C5
- Channel 2 - R2, R3, C3
- Channel 3 - R16, R17, C6
- Channel 4 - R10, R11, C4

2.3.3 Digital Circuitry

This section describes the digital circuitry around the device.

J2 and J3 are the main header pins that connect the digital and power pins to the SCB Controller or other EVMs. J3 connects to the EVM/SCB on the right, while J2 connects to more EVMs on the left. R5 and R6 are used as pullup resistors for the main I2C Bus.

SW0 and SW1 set the I2C address of the device. This can be useful when using the EVM with a custom controller (other than the SCB Controller), or when connecting multiple EVMs together. Currently the SCB Controller and GUI are set up to use four EVMs at a time.

R8 is used as a pullup resistor for the ALERT pin. LED D2 and current limiting resistor R7 are used to indicate when the ALERT has triggered. U2 is an open-drain buffer that forwards the alert signal to J2 and J3 without allowing the signal to propagate from the ALERT bus to the device. This feature is primarily used when working with multiple EVMs, so that the individual ALERT LEDs can be seen on each EVM while still using the ALERT bus. C8 is a bypass capacitor placed near the buffer to mitigate power supply noise and to help provide current quickly to the device when needed.

When connecting multiple EVMs together, if the pullup resistors for the I2C Bus or the ALERT Bus are pulling up too strongly, then remove the pullup resistors from some of the EVMs.

2.4 PAMB Compatibility

If desired, this EVM and GUI can be used with the PAMB Controller (DC081A) by jumper wiring the pin headers of the PAMB to the EVM. Figure 2-5 shows which pins on the PAMB correspond to the EVM header pins. Note not to add too much resistance in the jumper wire connection setup or the signal can degrade and cause communication errors.

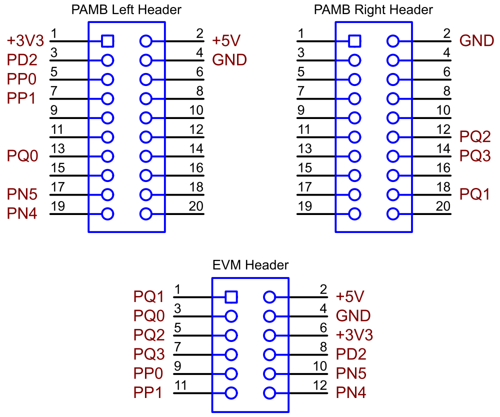


Figure 2-5. EVM to PAMB Connection

3 Software

3.1 Setup

3.1.1 Driver Installation

Download and install this driver: https://www.ti.com/lit/zip/sbac253. This is a one-time step per computer, and requires a myTI account. Note that this driver is labeled as a PAMB driver, but is also used for the SCB. Unzip the folder and run the .exe file with administrator privileges.

3.1.2 Firmware

Firmware updates are pushed through the GUI if the previous driver is installed. Downloaded offline GUIs only update the SCB Controller with the latest firmware available at the time of download. To check for the latest GUI or firmware updates, launch the latest GUI version from the web browser.

3.1.2.1 Firmware Debug

If the firmware must be manually reinstalled for any reason, then follow these steps to reinstall the firmware. Verify that the EVM is connected to the SCB before being powered on.

1. First, see if the GUI can program the firmware manually.
    a. Plug in the SCB controller to the PC.
        i. Verify that the EVM is connected to the SCB first.
    b. Launch the GUI and close the README.md window.
    c. If the MCU is already in Device Firmware Upgrade (DFU) mode, then a dialog box appears saying "Connection Failure - No SCB controller detected". Close this window.
        i. For a new SCB, the LEDs cycles 4 times and then enters DFU mode automatically.
    d. If the GUI does not update automatically, then go to *File > Program Device...*.
        i. If the *Program Device...* button is still grayed out, then select the connect button in the lower left corner and then try again.
2. If step 1 is unsuccessful (or if the *Program Device...* button is still grayed out), then manually configure the MCU on the SCB Controller to be in DFU Mode. This can be done through either of the below methods with the SCB Controller powered on:
    a. Through software:
        - Send the command 'bsl' on the USB Serial (COM) port of the SCB.

b. Through hardware:
   - For safety, **turn off and disconnect all load sources and external voltages**.
   - While shorting the two test points labeled *DFU* (shown in Figure 3-1) with a pair of tweezers (or wire), press and release the *RESET* button.

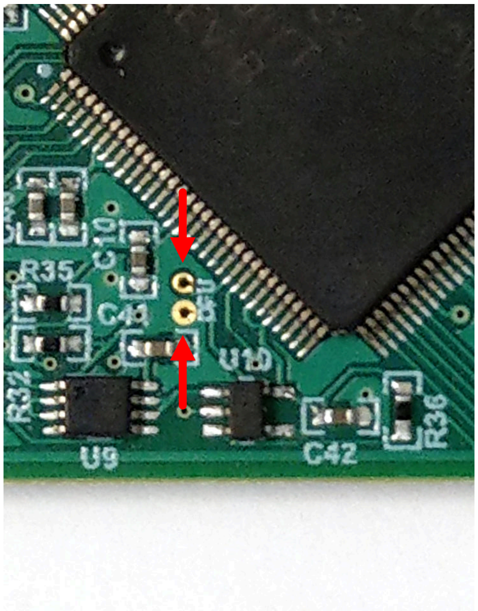


Figure 3-1. Test Points Used to Enter DFU Mode Manually

   - If the PAMB board is being used instead, then these test points are located near PK1 and PK2.

With the MCU in DFU mode, the firmware can now be uploaded through the method outlined in Step 1. If the device is in DFU mode, but the firmware update continues to fail, then manually associate the driver ("boot_usb.inf") with the DFU device.

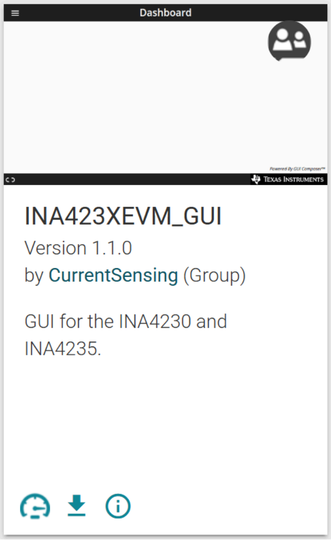
3.1.3 GUI Setup and Connection

You can access the GUI from this link in either Chrome®, Firefox®, or Safari®: https://dev.ti.com/gallery/info/CurrentSensing/INA423XEVM_GUI.

3.1.3.1 Initial Setup

To set up the GUI the first time:

1. Make sure that the previously mentioned driver is installed successfully so that everything works properly and that the GUI can update the EVM firmware, if necessary.
2. Check to make sure the EVM and SCB Controller unit is plugged into the PC, then go to the previously-provided GUI link.
   a. Verify that the EVM is connected to the SCB before plugging the SCB into a USB port.
3. Open the *GUI Composer* application to launch the GUI from the web browser (see Figure 3-2).



INA423XEVM_GUI

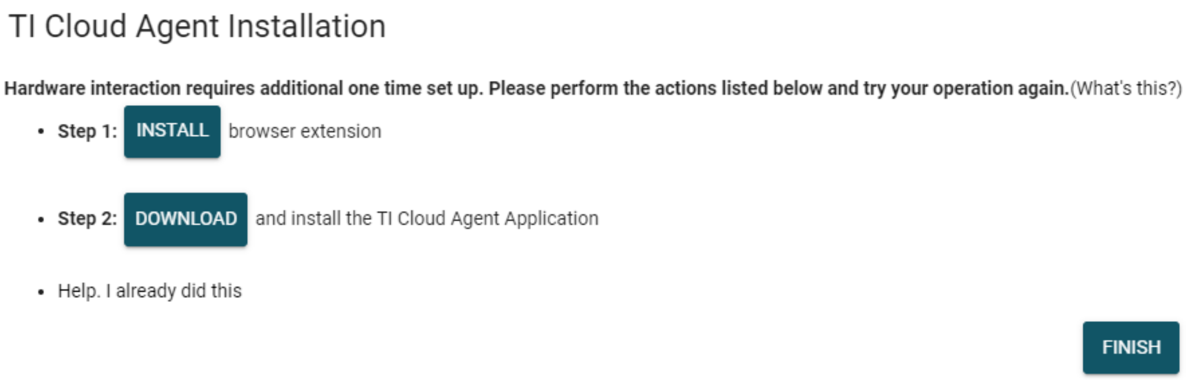
Version 1.1.0

by CurrentSensing (Group)

GUI for the INA4230 and INA4235.

Figure 3-2. GUI Composer Application

a. The GUI link brings up all versions of the GUI. TI recommends launching the newest version available.
b. For first-time GUI Composer setup, follow the prompts to download the *TI Cloud Agent* and browser extension (see Figure 3-3). These prompts appear after closing the *README.md* dialog box.



TI Cloud Agent Installation

Hardware interaction requires additional one time set up. Please perform the actions listed below and try your operation again.(What's this?)

- **Step 1:** INSTALL browser extension
- **Step 2:** DOWNLOAD and install the TI Cloud Agent Application
- Help. I already did this

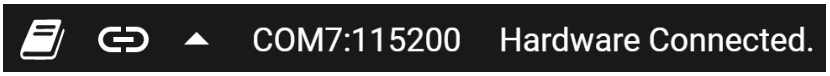
Figure 3-3. TI Cloud Agent

4. Optionally, to download the GUI for offline use, select the ⤓ icon in the *GUI Composer* application and follow the prompts (see Figure 3-2).

3.1.3.2 GUI to EVM Connection

To connect the GUI to the EVM, follow these steps:

1. Setup and launch the GUI as described in *Initial Setup*.
   a. Connect the EVM to the SCB before powering on.
2. Close the README.md file page to initiate a connection between the EVM and the GUI. If successful, then the text *Hardware Connected* is visible near the bottom left corner of the GUI.



COM7:115200 Hardware Connected.

Figure 3-4. Hardware Connected

   a. A green indicator with the device type and the text *DEVICE CONNECTED* is also visible near the top left of the GUI.

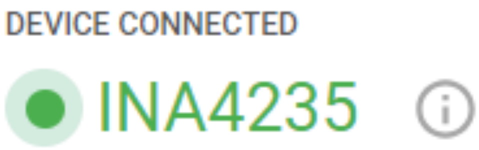


Figure 3-5. Device Connected

b. If *Hardware Connected* and *DEVICE CONNECTED* do not show in the GUI, then long-press the RESET button on the EVM to try again.
   i. If that option does not work, then check different hardware COM ports under *Options* > *Serial Port*.

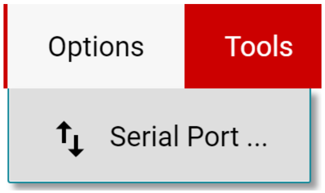


Figure 3-6. Change Serial Port

c. If the hardware still does not connect, then make sure the correct GUI and EVM combination is being used.
   i. If using the correct GUI and EVM combination, then reprogram the firmware of the SCB as described in *Firmware Debug*.
   ii. Many connectivity issues can be addressed by doing one of the following:
      1. Long-press the RESET button on the EVM with the EVM and SCB connected to each other.
         - Refreshing the GUI can also sometimes help this.
      2. Connect the EVM to a different USB port.
         - Avoid using long cables and USB hubs.
         - If using a desktop PC, try a USB port on the back.

3.2 GUI Operation

Setup, launch, and connect the GUI to the EVM per *GUI Setup and Connection*. Refer to the sections below for a description on how to use each page of the GUI.

3.2.1 Homepage Tab

The GUI starts out on the homepage tab. Click the ⌂ (Homepage) icon on the menu to the left to return to the homepage tab at any time.

From the homepage, you can easily confirm a successful GUI to EVM connection (see *GUI to EVM Connection*). There are also helpful resources available through the buttons on the bottom (see Figure 3-7).

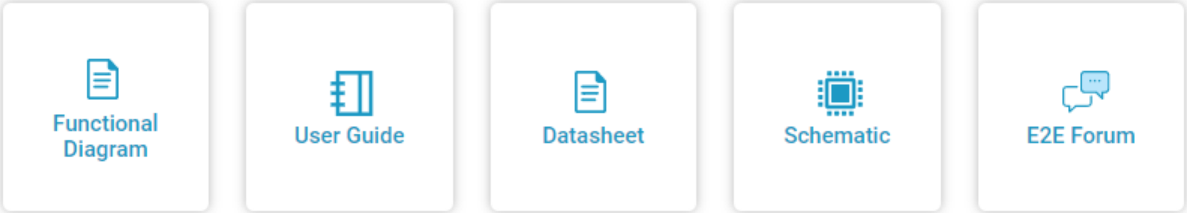
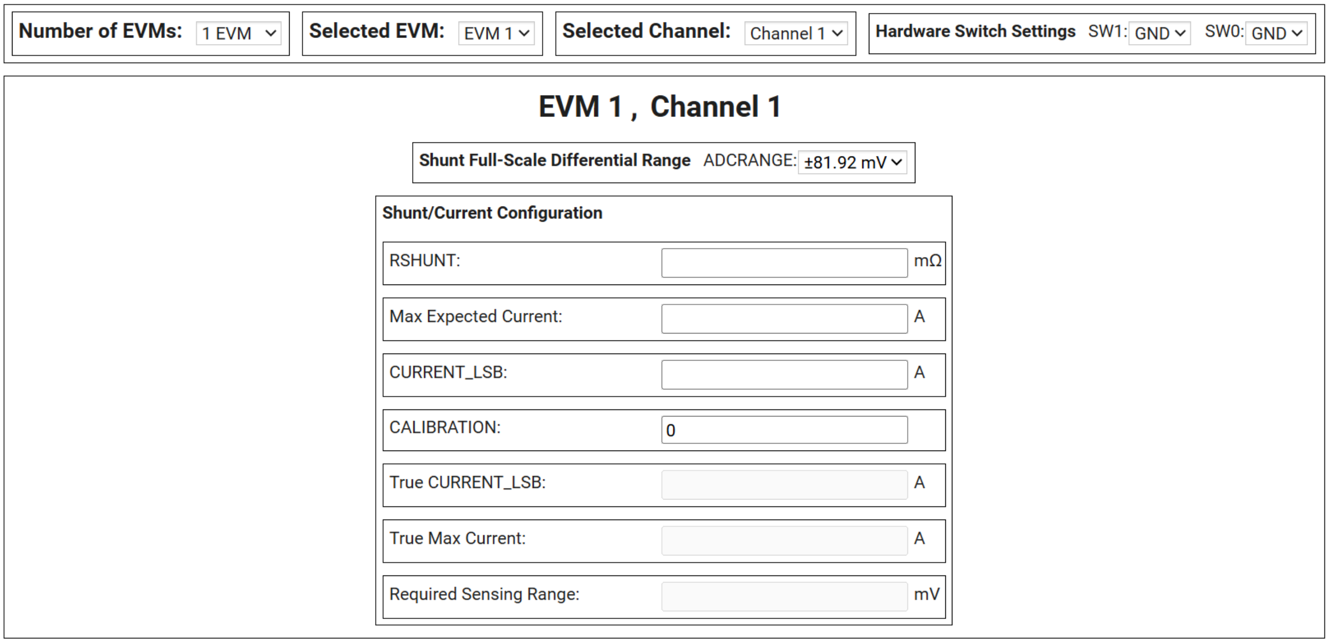


Figure 3-7. Home Tab Links

3.2.2 Configuration Tab

To do the initial setup for each connected EVM, click the ⚙ (*Configuration*) icon on the menu to the left. Figure 3-8 shows an example of the configuration tool.



Number of EVMs: 1 EVM

Selected EVM: EVM 1

Selected Channel: Channel 1

Hardware Switch Settings SW1: GND SW0: GND

EVM 1 , Channel 1

Shunt Full-Scale Differential Range ADCRANGE: ±81.92 mV

Shunt/Current Configuration

RSHUNT: mΩ

Max Expected Current: A

CURRENT_LSB: A

CALIBRATION: 0

True CURRENT_LSB: A

True Max Current: A

Required Sensing Range: mV

Figure 3-8. Configuration Tool

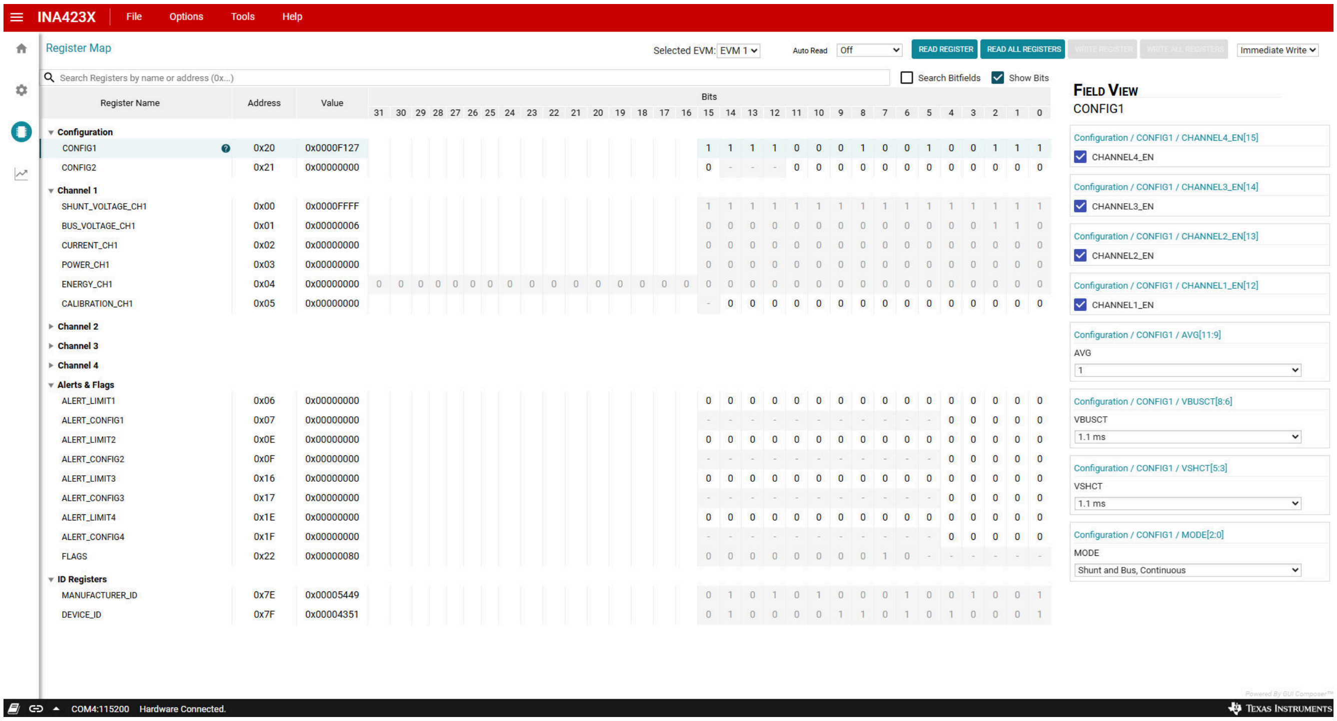
From this page, the user can set the number of EVMs to use, and then for each EVM, indicate the physical hardware switch settings and configure the shunt and CURRENT_LSB for each channel. A description of each option and field on this page is located below:

- Number of EVMs
  - This setting is used to tell the GUI how many EVMs are connected to the SCB.
  - Note, the SCB and GUI only support up to four EVMs at a time with the same device on each EVM.
  - Changing this here also changes the same setting in the *Results Data* tab.

- Selected EVM
  - This setting indicates which EVM the user is changing settings for.
  - This also selects the EVM that is connected to the *Registers* tab.
- Selected Channel
  - This setting indicates which channel the user is changing settings for.
- Hardware Switch Settings
  - Set these settings to match the physical switch settings on the EVM.
  - Note, this setting needs to be set before changing any other settings on this page. The GUI blocks the other settings until this is set.
  - Changing this here also changes the same setting in the *Results Data* tab.
- Shunt Full-Scale Differential Range
  - This is a shortcut to the ADCRANGE setting from the *Registers* tab; placed here for convenience.
  - Both this setting and the ADCRANGE setting from the *Registers* tab change together.
- Shunt and Current Configuration
  - This section is used to input shunt information as well as to help calculate the CURRENT_LSB and set the CALIBRATION register. Here is a description of how to use each field:
    - RSHUNT
      - Input the value of the used shunt resister in mΩ.
    - Max Expected Current
      - Input the value of the maximum expected current across the shunt resistor in Amps.
      - If the Max Expected Current field is left blank, then CALIBRATION can be adjusted manually, and the tool tells the True Max Current that can be measured with the EVM.
    - CURRENT_LSB
      - This is the calculated CURRENT_LSB value in Amps. This field gets populated automatically from the Max Expected Current field.
      - This field can be changed manually if desired, and changes filter downward.
    - CALIBRATION
      - Calculated value for CALIBRATION based off of RSHUNT and CURRENT_LSB. When this field changes, the value is automatically written to the CALIBRATION register.
      - When CURRENT_LSB is specified, changing ADCRANGE adjusts this value automatically per data sheet equations.
        - This happens if ADCRANGE is changed from either the *Configuration* tab or the *Registers* tab.
        - If CURRENT_LSB is not specified, then this value remains unchanged, but the fields below adjusts to the new ADCRANGE.
      - This field can be changed manually if desired, and changes filter downward.
      - Changing this value from the *Registers* tab also changes here.
    - True CURRENT_LSB
      - This is the actual CURRENT_LSB value in Amps back calculated from the CALIBRATION register with the given shunt resistor value.
      - This is the value used for calculations in the *Results Data* tab.
    - True Max Current
      - This is the maximum measurable current in Amps based off of the VSHUNT and CURRENT registers, using RSHUNT and the True CURRENT_LSB for calculations.
    - Required Sensing Range
      - This shows the required sensing range to measure the Max Expected Current with the specified shunt resistor.
      - If a Max Expected Current is not specified, then the True Max Current field is used instead.

3.2.3 Registers Tab

To view and edit the device registers, click the (*Registers*) icon on the menu to the left. The *Registers* tab looks similar to the one shown in Figure 3-9.



Register Map

Selected EVM: EVM 1

Auto Read: Off

Immediate Write

Search Registers by name or address (0x...)

☐ Search Bitfields ☑ Show Bits

Register Name	Address	Value	Bits																															
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Configuration																																		
CONFIG1	0x20	0x0000F127																	1	1	1	1	0	0	0	1	0	0	1	0	0	1	1	1
CONFIG2	0x21	0x00000000																	0	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0
Channel 1																																		
SHUNT_VOLTAGE_CH1	0x00	0x0000FFFF																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
BUS_VOLTAGE_CH1	0x01	0x00000006																	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
CURRENT_CH1	0x02	0x00000000																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
POWER_CH1	0x03	0x00000000																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ENERGY_CH1	0x04	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CALIBRATION_CH1	0x05	0x00000000																	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Channel 2																																		
Channel 3																																		
Channel 4																																		
Alerts & Flags																																		
ALERT_LIMIT1	0x06	0x00000000																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ALERT_CONFIG1	0x07	0x00000000																	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0
ALERT_LIMIT2	0x0E	0x00000000																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ALERT_CONFIG2	0x0F	0x00000000																	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0
ALERT_LIMIT3	0x16	0x00000000																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ALERT_CONFIG3	0x17	0x00000000																	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0
ALERT_LIMIT4	0x1E	0x00000000																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ALERT_CONFIG4	0x1F	0x00000000																	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0
FLAGS	0x22	0x00000080																	0	0	0	0	0	0	0	0	1	0	-	-	-	-	-	-
ID Registers																																		
MANUFACTURER_ID	0x7E	0x00005449																	0	1	0	1	0	1	0	0	0	1	0	0	1	0	0	1
DEVICE_ID	0x7F	0x00004351																	0	1	0	0	0	0	1	1	0	1	0	1	0	0	0	1

FIELD VIEW

CONFIG1

Configuration / CONFIG1 / CHANNEL4_EN[15]

☑ CHANNEL4_EN

Configuration / CONFIG1 / CHANNEL3_EN[14]

☑ CHANNEL3_EN

Configuration / CONFIG1 / CHANNEL2_EN[13]

☑ CHANNEL2_EN

Configuration / CONFIG1 / CHANNEL1_EN[12]

☑ CHANNEL1_EN

Configuration / CONFIG1 / AVG[11:9]

AVG: 1

Configuration / CONFIG1 / VBUSCT[8:6]

VBUSCT: 1.1 ms

Configuration / CONFIG1 / VSHCT[5:3]

VSHCT: 1.1 ms

Configuration / CONFIG1 / MODE[2:0]

MODE: Shunt and Bus, Continuous

COM4:115200 Hardware Connected.

Figure 3-9. GUI Registers Tab

From this page, the user can read and write device registers on the EVM. Here are some important notes:

- Use the *Selected EVM* drop-down menu at the top to choose which device to work with on the *Registers* tab.
  - Note, changing this here also changes the same setting for the *Configuration* tab.
  - Functionally, this button sets the default read/write address in the MCU and then reads all register values back to update the register map. Note that if data is currently collecting at a high frequency, then this can cause a minor delay in the data collection. To prevent this, set the device settings before starting data collection.
- By default, all changes are automatically written to the device. If desired, then the user can change the *Immediate Write* setting to *Deferred Write* to only allow writing when ready.
  - Users can modify writable register values from any of these methods:
    - Through the widget settings in the *Field View* section on the right.
    - Changing the *Value* directly with either hex or decimal values.
    - Double-clicking on any individual bit.
- Turning on *Auto Read* only updates registers in the *Registers* tab, and not the plots in the *Results Data* tab.
  - Leaving *Auto Read* on while collecting data for plots can interfere with data collection timing.
- For questions about a register or register bit field, click the ? icon.
  - For even more questions about registers, check the data sheet.
- For convenience, register settings can be saved and loaded back later to any device with the same register map. To do this, go to *File* > *Register Data*, as shown in Figure 3-10.

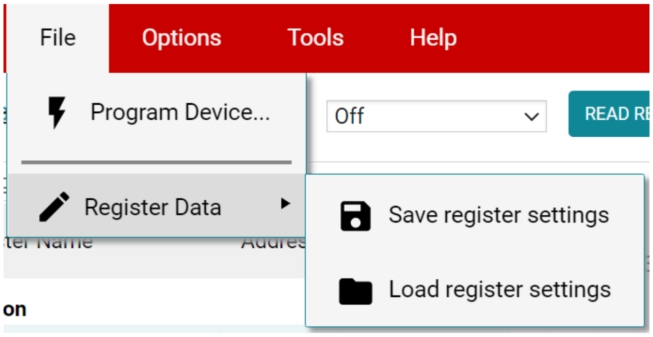
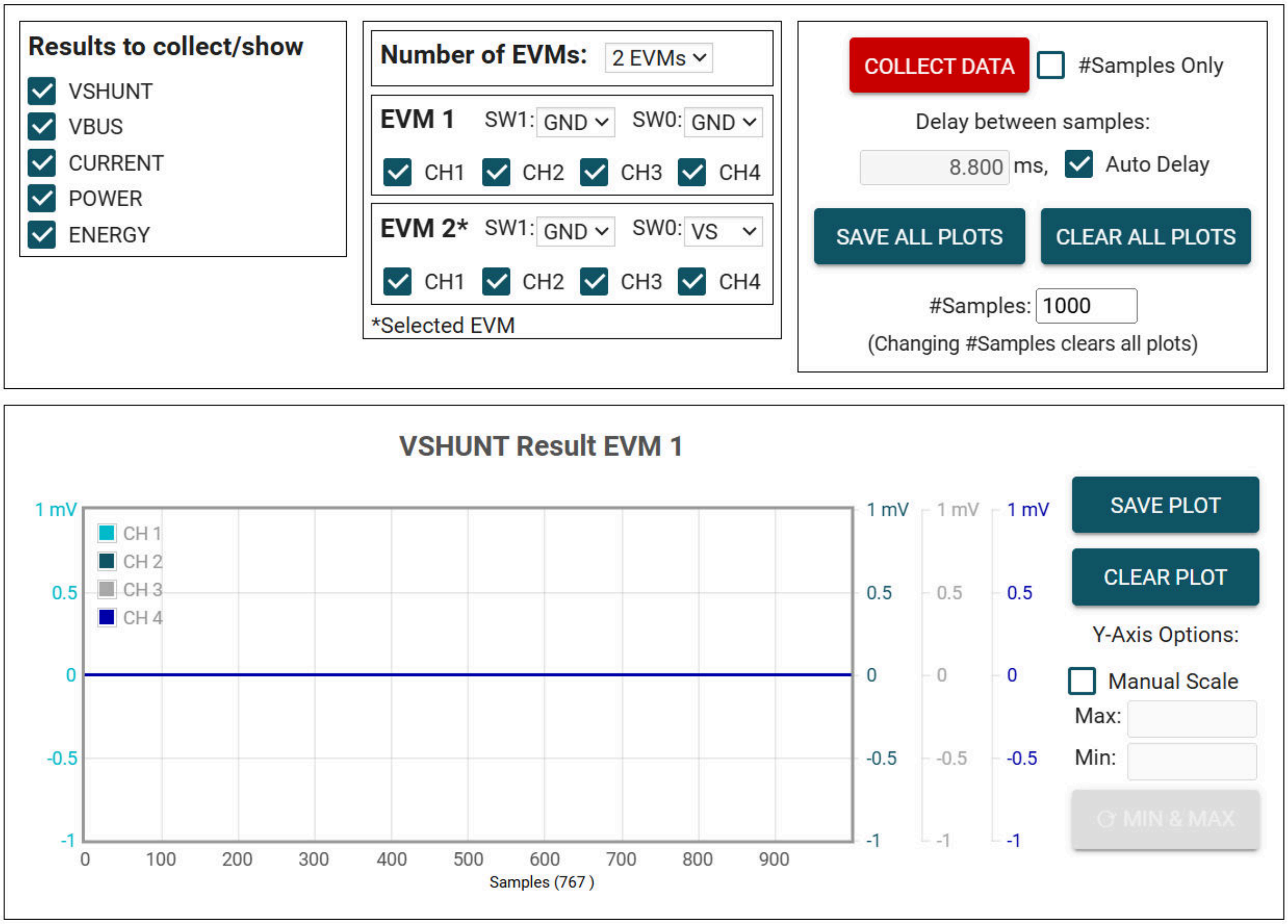


Figure 3-10. Save and Load Register Settings

– Press the *Read All Registers* button after loading data to update the *Registers* tab with the actual device values, in case the loaded registers are not compatible with the connected device.

3.2.4 Results Data Tab

To view and collect results data over time, click the (Results Data) icon on the menu to the left. Figure 3-11 shows part of the *Results Data* tab for reference, which can look different depending on the number of connected EVMs.


Results to collect/show
☑ VSHUNT
☑ VBUS
☑ CURRENT
☑ POWER
☑ ENERGY

Number of EVMs: 2 EVMs

EVM 1 SW1: GND SW0: GND
☑ CH1 ☑ CH2 ☑ CH3 ☑ CH4

EVM 2* SW1: GND SW0: VS
☑ CH1 ☑ CH2 ☑ CH3 ☑ CH4

*Selected EVM

□ #Samples Only

Delay between samples: 8.800 ms, ☑ Auto Delay

#Samples: 1000

(Changing #Samples clears all plots)

□ Manual Scale
Max:
Min:

Figure 3-11. Results Data Tab and Settings

A description of how to use the buttons and settings at the top of the *Results Data* tab and next to each plot is below:

- Results to collect/show

  - Use this section to select which register values to collect data for. If a results register is unselected before the *COLLECT DATA* button is pressed, then the plot below is hidden and the EVM does not try to read this register during the collect cycle (even if the conversion is enabled).
  - If the user disables one of these settings while the EVM is collecting data, then the plot does not show, but data is still collected and the plot updates in the background. Reselect to show data.
- Number of EVMs
  - Set the *Number of EVMs* drop-down menu to the number of EVMs currently in use.
    - See Figure 2-2 for how to attach multiple EVMs together.
    - Changing this here also changes the same setting in the Configuration tab.
  - The GUI only supports one EVM and device type at a time, up to four EVMs total.
- EVM settings
  - Switch settings
    - Use the onboard switches to select a different address for each EVM.
    - Set the switch settings in the GUI to match the setting for each connected EVM.
      - *EVM 1* automatically populates with the lowest addressed device found when the SCB is powered up, unless a setting has already been selected.
      - Changing this here also changes the same setting in the *Configuration* tab.
    - If more than one device is being used, then an asterisk symbol (*) appears next to the selected EVM that is being used on the *Registers* and *Configuration* tabs.
      - Changing the switch settings of any EVM sets that EVM as the selected EVM.
  - Channel checkboxes
    - Use this section to select which channels to collect data for. If a channel is unselected before the *COLLECT DATA* button is pressed, then the plot below is hidden and the EVM does not try to read any results from that channel during the collect cycle (even if the channel conversion is enabled).
    - If the user disables one of these settings while the EVM is collecting data, then the plot does not show, but data is still collected and the plot updates in the background. Reselect to show data.
- Collect and Plots settings
  - Collect Data
    - Press the *COLLECT DATA* button to start data collection.
    - In this mode, the MCU reads and sends the selected result values for each device over a USB BULK channel. All results from one device are read before moving on to the next device.
      - All result values from all EVMs together are considered one *sample set*.
      - Currently all channels are collected per result.
    - Although users can read and write to other registers through the *Registers* tab while collecting data,this can add a delay to the data being collected.
    - Press the *STOP COLLECT* button to stop collecting data.
  - #Samples Only
    - If this checkbox is selected, then the GUI automatically stops collecting data after the number of samples specified in the #Samples box have been collected.
    - If not selected, then the GUI continues collecting data and only stores the most recent #Samples.
  - Delay between samples
    - Sets the delay between the start of each sample set.
    - Desired delay time is not obtainable if set faster than the read loop, which depends on the number of results being collected, the number of EVMs, and the CPU.
    - Although users can read and write to other registers through the *Registers* tab while collecting data, this can add a delay to the data being collected.
  - Auto Delay
    - Sets delay based on conversion times, averaging, and number of channels being converted.
    - If multiple EVMs are being used, then the time put in the delay box is from the EVM with the shortest calculated delay value.
  - Save All Plots
    - Press the *SAVE ALL PLOTS* button to save the data for each currently visible plot to a spreadsheet.
      - The visible plots change based on the selected result from the *Results to collect/show* section and the channel selection boxes for each EVM.

    - Press the *SAVE PLOT* button next to each plot to save just the data from that plot in a spreadsheet.
  - Clear All Plots
    - Press the *CLEAR ALL PLOTS* button to clear the data from all plots together.
    - Press the *CLEAR PLOT* button next to each plot to clear the data from just that plot.
  - #Samples
    - Change the number in this box to change the number of samples shown in each plot.
      - Changing this number clears out the plot buffers, so the plots are cleared on the next read.
  - Y-Axis Options
    - Manual Scale
      - Checking this box sets all channel results in this plot to the same scale value specified by the Max and Min fields.
      - When this is not selected, each channel has a Y-axis scale based on the min and max value for that channel's result values.
    - Max
      - The maximum Y-axis value to use for all channels in this plot.
      - If this field is empty when *Manual Scale* is selected, then the field auto-populates with the maximum value currently in the plot.
    - Min
      - The minimum Y-axis value to use for all channels in this plot.
      - If this field is empty when *Manual Scale* is selected, then the field auto-populates with the minimum value currently in the plot.
    - □ MIN & MAX
      - Press the □ *MIN & MAX* button to refresh the values in the *MIN* and *MAX* boxes based on the data currently visible in the plot.

3.3 Direct EVM USB Communication

If desired, the EVM can be communicated with directly without the use of the GUI through the USB port. This is done by sending the desired command string over the serial COM port and receiving the results either through the COM port or the USB BULK channel, based on the mode. This is useful for interfacing the EVM with custom setups, scripts, or GUIs.

3.3.1 Standard USB Read and Write Operations

Use the serial COM port to read and write registers through USB commands using the following format:

- Set device address format: setdevice DEVID
  - Where setdevice is always lower case, and DEVID is defined as:
    - The 4 LSBs of the device address in decimal format. For example, an address of 0x4A is 10.
  - Note, when the SCB is reset while one or more EVMs are connected, the address defaults to the lowest address found.
    - The SCB checks for I2C or SPI at start up. If no device is attached, then the device defaults to SPI. Reset the SCB with an I2C EVM connected to use I2C.
  - For example, to set the INA423XEVM with a register address of 0x4A, send the command: setdevice 10
  - For this example, the EVM returns the acknowledgment and state (*idle* or *collecting*) in JSON format:
    {"acknowledge":"setdevice 10"}
    {"evm_state":"idle"}
- Get device address format: getdevice
  - Where getdevice is always lower case.
  - Returns the current device address in decimal format.
  - For example, to get the address currently configured in the SCB, send the command: getdevice
  - For this example, if the address configured in the SCB is 0x4A, then the EVM returns the acknowledgment, the address, and state (*idle* or *collecting*) in JSON format:
    {"acknowledge":"getdevice"}
    {"Address":74}
    {"evm_state":"idle"}

- Read register format: rreg ADR
  - Where ADR is the address in hex, and rreg is always lower case.
  - Register addresses can be in upper or lower case, and does not need to be led by '0x'. 0 padding register addresses is also optional. For example, to read register address 0x0x7E, some valid commands include:
    - rreg 7e
    - rreg 07E
    - rreg 0x07E
      - When '0x' is used, the 'x' must be lower case.
  - For this example, the EVM returns the results and state (*idle* or *collecting*) in JSON format:
    {"acknowledge":"rreg 0x7E"}
    {"register":{"address":126,"value":21577}}
    {"evm_state":"idle"}
- Write register format: wreg ADR VAL
  - Where ADR and VAL are in hex, and wreg is always lower case.
  - Register addresses and values can be in upper or lower case, and does not need to be led by '0x'. 0 padding register addresses and values is also optional. For example, to write register address 0x1 with the value 0xfb69, some valid commands include:
    - wreg 20 f16f
    - wreg 20 0xf16f
    - wreg 0x20 0xF16F
      - When '0x' is used, the 'x' must be lower case.
  - For this example, the EVM returns the results and state (*idle* or *collecting*) in JSON format:
    {"acknowledge":"wreg 0x20 0xF16F"}
    {"console":"Writing 0xF16F to CONFIG1 register"}
    {"evm_state":"idle"}

3.3.2 Collect Data Through the USB BULK Channel

The *Collect Data* function reads the desired result registers and sends the data based on the specified settings. This function works best with continuous conversion mode and does not configure the EVM or associated register settings. Collect mode starts and stops via the serial COM port, however the results are sent over the USB BULK channel. To use this mode, use the following format:

- Start collecting data format: collect timerPeriod collectFlags channelAddressIDs numDevices
  - Where collect is always lower case, and each parameter is the decimal representation of the value in the following format:
    - timerPeriod
      - The timer delay used in the MCU to allow data collection sample sets (in µs, unsigned 32-bit value).
    - collectFlags (All channels collected together)
      - a byte of data that has a 1 to collect and a 0 to not collect each register value type, according to the following definitions:
        - VSHUNT = 0b10000
        - VBUS = 0b01000
        - CURRENT = 0b00100
        - POWER = 0b00010
        - ENERGY = 0b00001
    - channelAddressIDs
      - This is the 4 LSBs of each I2C address chained together, starting with bits 3 - 0.
        - For example, if EVM 1 is on address 0x41 and EVM 2 is on 0x43, the value here is 0b00110001
    - NumDevices
      - The number of EVMs chained together (1-4).
  - For example, to start data collection for VSHUNT, VBUS, and POWER every 8.8ms, for two INA4235s with EVM 1 address = 0x41 and EVM 2 address = 0x43, the user needs to send: collect 8800 26 49 2
    - For this example, the EVM returns the acknowledgment and state in JSON format:
      {"acknowledge":"collect 8800 26 49 2"}
      {"evm_state":"collecting"}
  - The USB BULK channel receives data in the format: frameID deviceNumID address registerSize data
    - Where each parameter is the decimal representation of the value in the following format:
      - frameID (1 byte)
        - Always reads 0. Used to verify data is aligned.
      - deviceNumID (1 byte)
        - An ID number corresponding to the EVM number.
          - From the above example, this is 1 if reading from EVM 1, which had an address set to 0x41, and 2 if reading from EVM 2 which had an address set to 0x43.
      - address (1 byte)
        - The register address that is read from the device.
      - registerSize (1 byte)
        - The number of bytes that the following data has.
      - data (1 byte at a time)
        - The register data value, given in bytes with the most significant byte first.
- Stop collecting data format: stop
  - Where stop is always lower case.
  - The EVM returns the acknowledgment and state in JSON format:
    {"acknowledge":"stop"}
    {"evm_state":"idle"}

4 Hardware Design Files

4.1 Schematics

Figure 4-1 and Figure 4-2 show the schematic of the EVM. Figure 4-1 shows the circuitry for the EVM. Figure 4-2 shows the mechanical components included with the EVM.

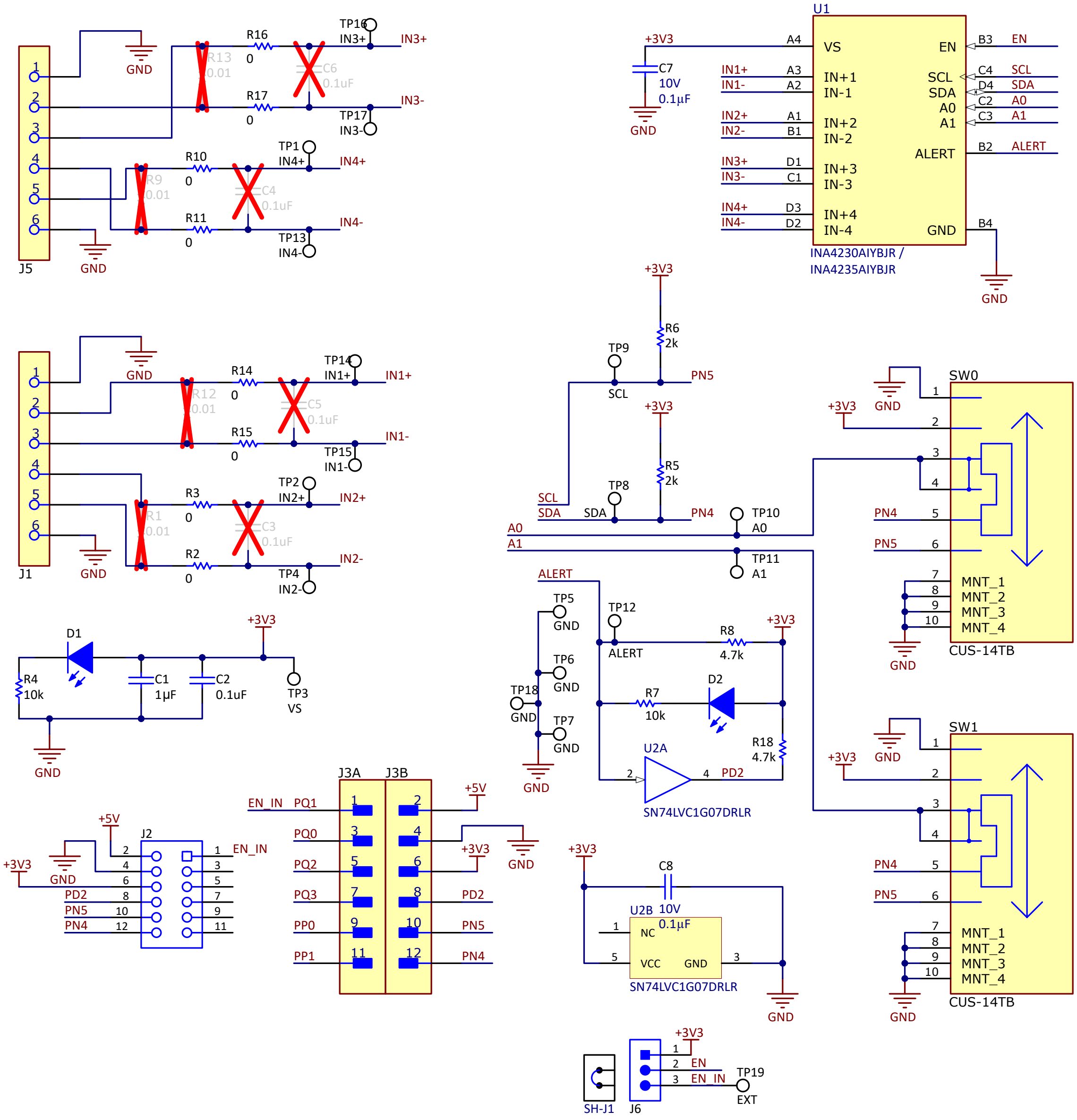
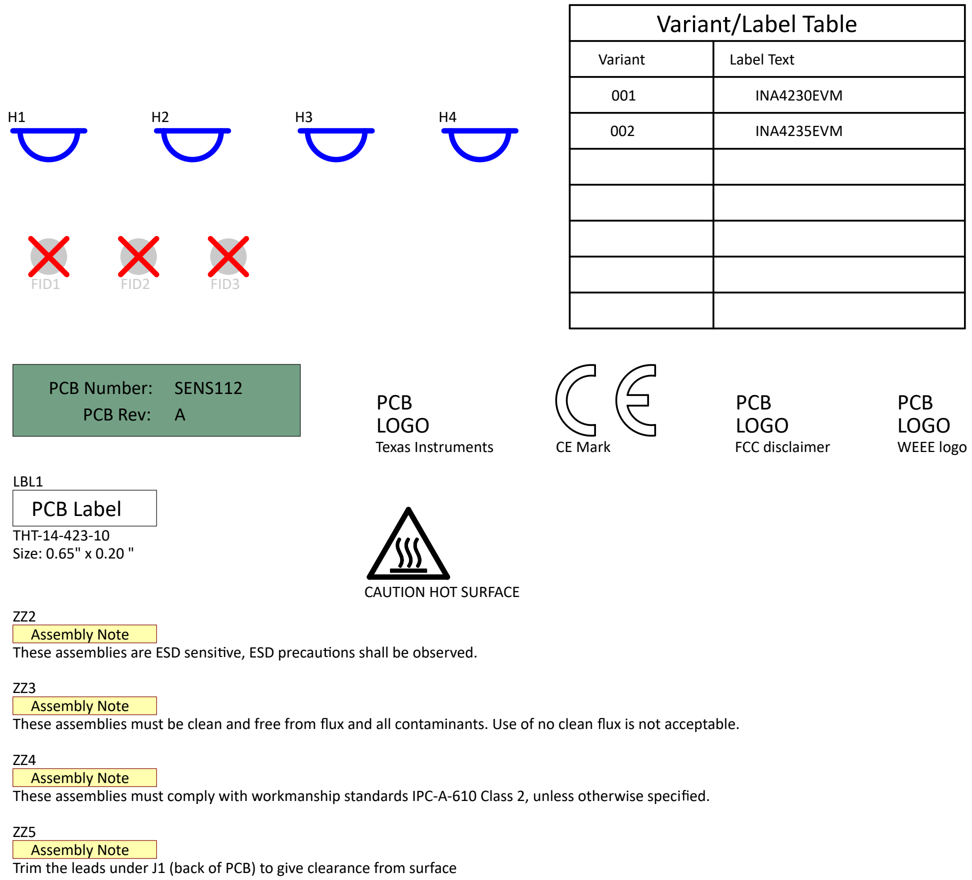


Figure 4-1. SENS112 Schematic



Variant/Label Table

Variant	Label Text
001	INA4230EVM
002	INA4235EVM

Assembly Note
These assemblies are ESD sensitive, ESD precautions shall be observed.

Assembly Note
These assemblies must be clean and free from flux and all contaminants. Use of no clean flux is not acceptable.

Assembly Note
These assemblies must comply with workmanship standards IPC-A-610 Class 2, unless otherwise specified.

Assembly Note
Trim the leads under J1 (back of PCB) to give clearance from surface

Figure 4-2. SENS112 Hardware Schematic

4.2 PCB Layout

Note

Board layouts are not to scale. These figures are intended to show how the board is laid out. The figures are not intended to be used for manufacturing EVM PCBs.

Figure 4-3 through Figure 4-6 illustrate the PCB layers of the EVM.

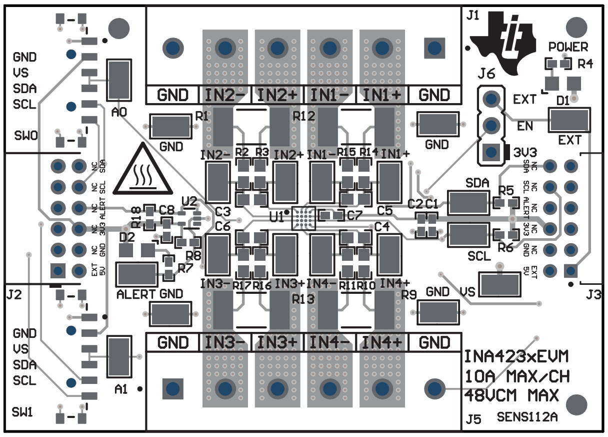


Figure 4-3. SENS112 Top View

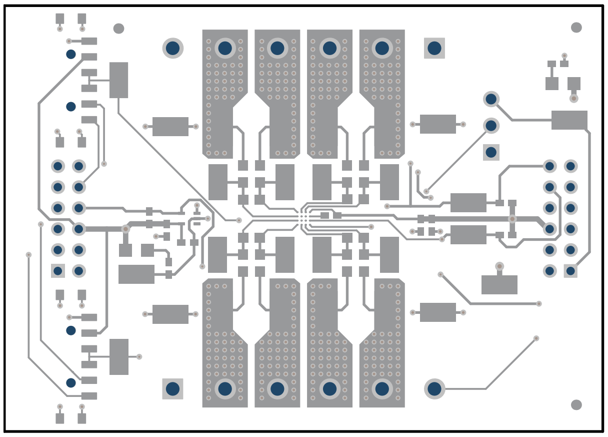


Figure 4-4. SENS112 Top Layer

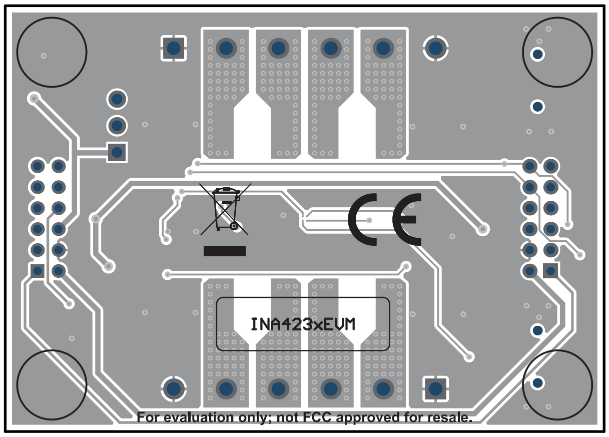


Figure 4-5. SENS112 Bottom View

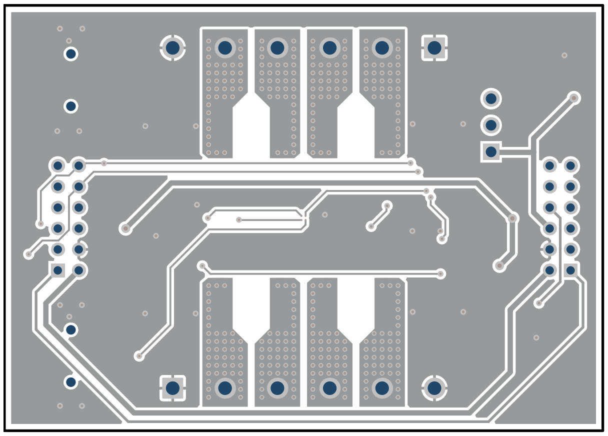


Figure 4-6. SENS112 Bottom Layer

4.3 Bill of Materials

Table 4-1 through Table 4-3 provide the parts list for the EVM. Table 4-1 through Table 4-2 are variant specific, while Table 4-3 shows the parts common to all SENS112 variants.

Table 4-1. INA4230EVM Exclusive Bill of Materials

Designator	QTY	Value	Description	Package Reference	Part Number	Manufacturer
U1	1		48V, Quad Channel, 16-Bit, Precise, Current, Voltage, Power, and Energy Monitor With an I2C Interface	DSBGA16	INA4230AIYBJR	Texas Instruments

Table 4-2. INA4235EVM Exclusive Bill of Materials

Designator	QTY	Value	Description	Package Reference	Part Number	Manufacturer
U1	1		48V, Quad Channel, 16-Bit, Precise, Current, Voltage, Power, and Energy Monitor With an I2C Interface	DSBGA16	INA4235AIYBJR	Texas Instruments

Table 4-3. SENS112 Variants Bill of Materials

Designator	QTY	Value	Description	Package Reference	Part Number	Manufacturer
C1	1	1uF	CAP, CERM, 1µF, 16V,+/- 20%, X5R, 0402	0402	GRM155R61C105MA12D	MuRata
C2	1	0.1uF	CAP, CERM, 0.1uF, 50V, +/- 20%, X7R, 0402	0402	GRM155R71H104ME14D	MuRata
C7, C8	2	0.1uF	CAP, CERM, 0.1µF, 10V,+/- 10%, X7R, AEC-Q200 Grade 1, 0402	0402	C0402C104K8RACAUTO	Kemet
D1, D2	2	White	LED, White, SMD	0805	VAOL-S8WR4	Visual Communications Company, LLC
H1, H2, H3, H4	4		Bumpon, Hemisphere, 0.25 X 0.075, Clear	75x250 mil	SJ5382	3M
J1, J5	2		TERM BLK 6POS SIDE ENTRY 5MM PCB ASSEMBLY NOTE: Trim leads per ZZ5	HDR6	6.91138E+11	Wurth Electronics
J2	1		Receptacle, 2mm, 6x2, Gold, R/A, TH	Receptacle, 2mm, 6x2, R/A, TH	NPPN062FJFN-RC	Sullins Connector Solutions
J3	1		Connector Header Through Hole, Right Angle 12 position 0.079" (2.00mm)	HDR12	NRPN062PARN-RC	Sullins Connector Solutions
J6	1		Header, 2.54mm, 3x1, Gold, TH	Header, 2.54mm, 3x1, TH	GBC03SAAN	Sullins Connector Solutions
LBL1	1		Thermal Transfer Printable Labels, 0.650" W x 0.200" H - 10,000 per roll	PCB Label 0.650 x 0.200 inch	THT-14-423-10	Brady
R2, R3, R10, R11, R14, R15, R16, R17	8	0	RES, 0, 5%, 0.125 W, 0603	0603	MCT06030Z0000ZP500	Vishay/Beyschlag
R4, R7	2	10k	RES, 10 k, 5%, 0.063 W, AEC-Q200 Grade 0, 0402	0402	CRCW040210K0JNED	Vishay-Dale
R5, R6, R8, R18	4	4.7k	RES, 4.7 k, 5%, 0.063 W, AEC-Q200 Grade 0, 0402	0402	CRCW04024K70JNED	Vishay-Dale
SH-J1	1		Shunt, 2.54mm, Gold, Black	Shunt, 2.54mm, Black	60900213421	Wurth Elektronik
SW0, SW1	2		Slide Switch SP4T Surface Mount, Right Angle	SMT_SW_11MM3_4MM1	CUS-14TB	Nidec Copal Electronics

Table 4-3. SENS112 Variants Bill of Materials (continued)

Designator	QTY	Value	Description	Package Reference	Part Number	Manufacturer
TP1, TP2, TP3, TP4, TP5, TP6, TP7, TP8, TP9, TP10, TP11, TP12, TP13, TP14, TP15, TP16, TP17, TP18, TP19	19		Test Point, Miniature, SMT	Testpoint_Keystone_Miniature	5015	Keystone
U2	1		Single Buffer/Driver With Open-Drain Output, DRL0005A, LARGE T&R	DRL0005A	SN74LVC1G07DRLR	Texas Instruments
C3, C4, C5, C6	0	0.1uF	CAP, CERM, 0.1uF, 100V, +/- 10%, X7S, AEC-Q200 Grade 1, 0603	0603	CGA3E3X7S2A104K080AB	TDK
FID1, FID2, FID3	0		Fiducial mark. There is nothing to buy or mount.	N/A	N/A	N/A
R1, R9, R12, R13	0		10 mOhms ±0.5% 2W Chip Resistor 2512 (6432 Metric) Automotive AEC-Q200, Current Sense, Moisture Resistant Metal Film	2512	PCS2512DR0100ET	Ohmite

5 Additional Information

5.1 Trademarks

Chrome® is a registered trademark of Google LLC.
Firefox® is a registered trademark of Mozilla Foundation.
Safari® is a registered trademark of Apple Inc.
All trademarks are the property of their respective owners.

6 Related Documentation

This user's guide is available from the TI website under literature number SBOU298. Any letter appended to the literature number corresponds to the document revision that is current at the time of the writing of this document. Newer revisions are available from www.ti.com or the Texas Instruments' Literature Response Center at (800) 477-8924 or the Product Information Center at (972) 644-5580. When ordering, identify the document by both title and literature number. Table 6-1 lists documentation related to the EVM. Click the links in Table 6-1 for further information. The device name links to the product web folder on www.ti.com. The literature number links to the document PDF.

Table 6-1. Related Documentation

DOCUMENT TITLE	DOCUMENT LITERATURE NUMBER
INA4230 data sheet	SBOSAD4
INA4235 data sheet	SBOSAB5
Getting Started with Digital Power Monitors	SBOA511

7 Revision History

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

Changes from Revision * (March 2024) to Revision A (February 2026) **Page**

- Updated the numbering format for tables, figures, and cross-references throughout the document 1
- Updated the entire document to support new GUI features.. 1
- Updated the *Device information* section..2
- Added clarifications and updates throughout the *Firmware* section...7
- Added *Direct EVM Serial Communication* section to explain how to communicate with the EVM via direct serial communication.. 16

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), 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 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, TI’s General Quality Guidelines, 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. Unless TI explicitly designates a product as custom or customer-specified, TI products are standard, catalog, general purpose devices.

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

Last updated 10/2025