

High-Voltage Solar Inverter DC-AC Kit

VieriXue

ABSTRACT

Inverters have gained a lot of attention in recent years, especially solar inverters. The solar inverter has solar energy input that feeds energy into the grid, therefore, grid-tie technology and protection are the key points when designing a solar inverter system.

This application report describes the implementation of the inverter kit that is used as a DC-AC part of the high-voltage solar kit. The kit has a nominal input of 400 V DC and its output is 600W, which can be fed to the grid. The following information is discussed in this document:

- Basic knowledge of the inverter
- Introduction of the kit
- Hardware introduction
- Firmware design
- Closed loop controllers design
- Build steps are introduced
- Test results and the waveform are shown

Contents

1	Introduction	2
2	Design Introduction	5
3	How to Build the Firmware	13
4	Test Result	20

List of Figures

1	Full Bridge Current Type Inverter	3
2	Single Polar Modulation Theory.....	4
3	The Typical Solar Inverter Structure.....	4
4	The Controller Loop of the Inverter Part in Solar System	5
5	The Key Components on the Board	6
6	The PCB Placement	6
7	Zero Crossing	8
8	IGBT Driver Diagram	8
9	The Firmware Structure.....	10
10	Status Machine	11
11	ADCDRV_5CH Block	13
12	The GEN_SIN_COS: n	14
13	INV_ICMD:n.....	15
14	PWMDRV:n	15
15	Change the Incremental Build	17
16	Debug Active Project.....	17
17	Open Loop Build.....	18

All trademarks are the property of their respective owners.

18	Close Loop Build Without PLL	18
19	The Connection of the Test	19
20	Close Loop With PLL Build.....	19
21	PF and THDi	22
22	The Efficiency	23
23	The System Structure and the Connection	23
24	The Turn on Overview	23
25	The DC-AC Turn on the PWM.....	24

List of Tables

1	The Key Points	7
2	The Jumper Setting for the Board	7
3	The LED Flashing Definition	12
4	C Files in Project	13
5	asm Files	13
6	Other Files.....	13
7	Incremental Build Option	16
8	The PF and THDi.....	21
9	Efficiency	22

1 Introduction

The inverter has been widely used in many fields, such as the motor control, the UPS, and the solar inverter systems. The main function of the inverter is to convert the DC power to AC power by using the power electronics like the IGBT, and MOSFET. Traditionally, many inverter systems will be implemented by the analog components. As the development of the digital processors, more and more low-cost and high-performance microcontrollers have gotten into the market. At the same time, more and more inverter systems tend to use the microcontrollers to implement the digital controller, which cannot only simplify the system structure but also improve the output performance of the inverters.

There are two different types among inverter systems. The first type is the voltage output type that outputs the AC voltage as a voltage source. For example, the inverter in the UPS system is a typical voltage type inverter. The other type is the current type, which outputs the AC current in a specified power factor. The motor control inverter and the solar inverter are the current type inverters. This document mainly discusses the current type inverter for the solar system.

At present, many different topologies for inverters have come onto the market. This demo uses the full-bridge topology (including four insulated-gate bipolar transistors (IGBT) as a reference design), which are easy to get started and transplant to the real product.

1.1 The Basic Principles

The full bridge current type inverter topology is shown in [Figure 1](#).

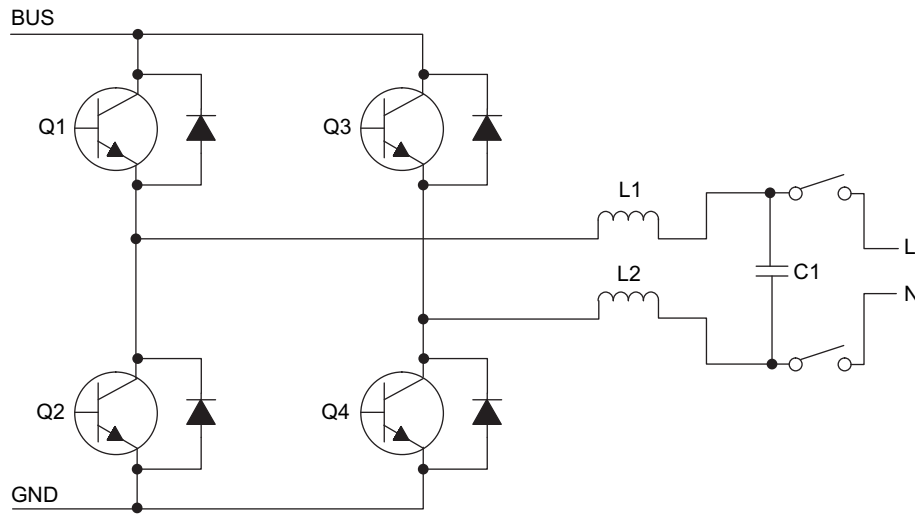


Figure 1. Full Bridge Current Type Inverter

There are many sinusoid pulse width modulation (SPWM) control strategies for the full bridge topology to have an AC output. Among these strategies, they can be divided into the following two categories:

- Single polar modulation
- Dual polar modulation

1.1.1 Single Polar Modulation Theory

The single polar means the voltage in the AC side of the inverter has only positive or only negative voltage. An example of the single polar modulation is shown in Figure 2.

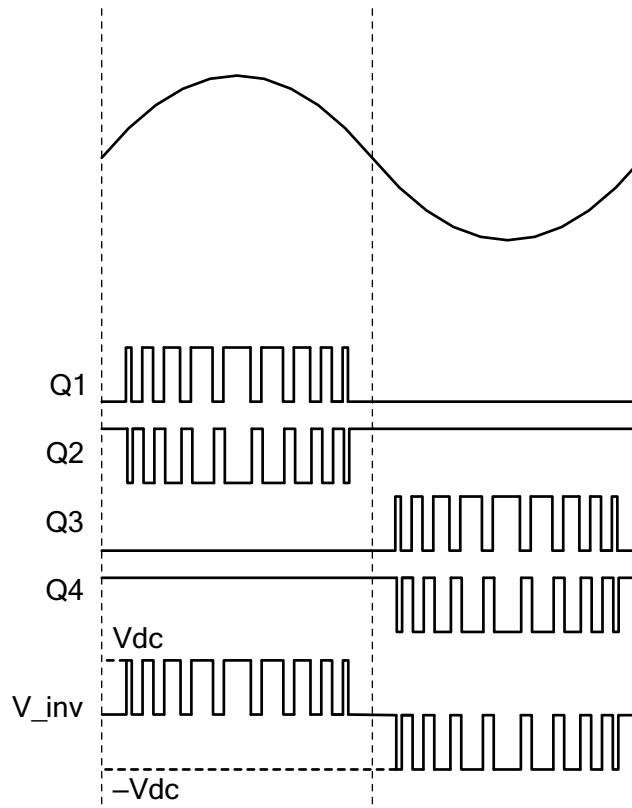


Figure 2. Single Polar Modulation Theory

Figure 2 shows that when in the positive cycle of the sine wave, the output voltage of the inverter is changing from the V_{dc} to 0, while the negative cycle is the $-V_{dc}$ to 0. So in the positive cycle, if the duty of Q1 is d , then you can get the relation between the output voltage V_o and the DC bus voltage V_{bus} :

$$V_o = dV_{bus} \quad (1)$$

1.1.2 The Controller Loop

For the current type inverter, the output current is controlled. Besides, in most of the solar inverter systems, there is a DC-DC part in front of the DC-AC part, which is used to boost up the panel voltage and execute the MPPT. The DC-DC will not control the DC bus voltage but controls the input panel voltage and works in the power output mode. So it is the responsibility for the DC-AC part (inverter) to control the DC bus voltage.

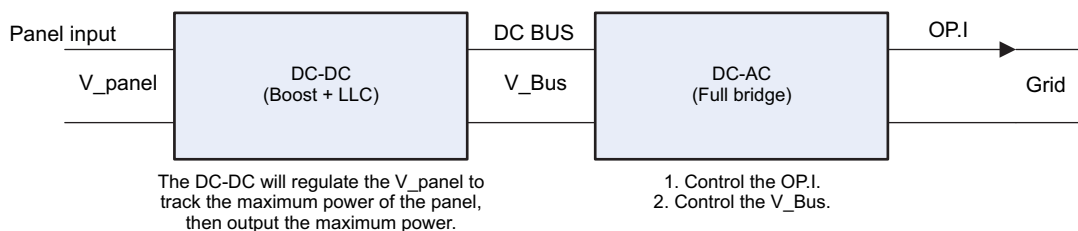


Figure 3. The Typical Solar Inverter Structure

The DC BUS works as a link between the DC-DC and DC-AC part. When the DC BUS voltage rises, the DC-AC increases its output current to keep the DC BUS at a specified value so that the output power of the system will be increased. When the DC BUS tends to fall down, the DC-AC decreases its output current to prevent the DC BUS from falling down, which will decrease the output power.

The typical controller structure for the inverter part is shown in Figure 4.

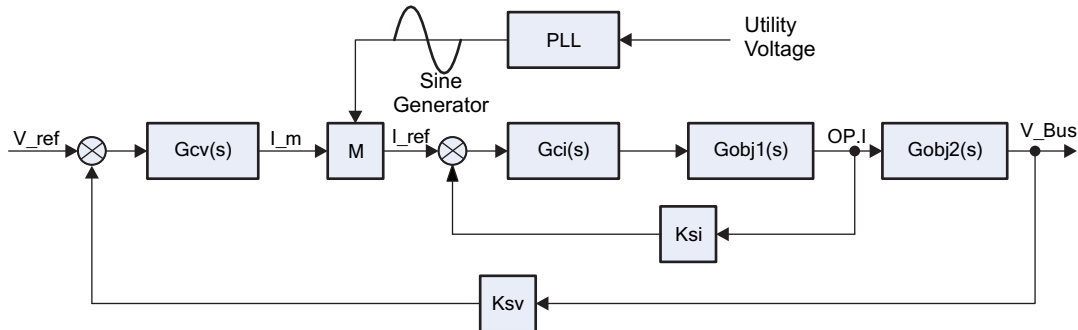


Figure 4. The Controller Loop of the Inverter Part in Solar System

The double loop control system is used in Fig 1.4. The internal loop is the output current loop, it will trace the I_{ref} which is the product of the I_m and Sine. The external loop is the DC BUS voltage loop, it will keep the bus voltage to V_{ref} . Besides, there is a PLL to ensure the synchronization of the grid voltage and the output current. Notes:

NOTE: When there is no DC-DC part and the DC-source in CV mode is connected, the external loop must be disabled.

2 Design Introduction

2.1 Hardware

2.1.1 The Key Component

The key components of the kit are shown in Figure 5. The following hardware is used in this kit:

- Four pieces of 600 V IGBT
- IGBT drivers are designed to the module type
- Two pieces of 2.5 mH inductor
- Two pieces of relay are used to control the grid-tie connection
- Hall current sensor is used to sense the inductor current

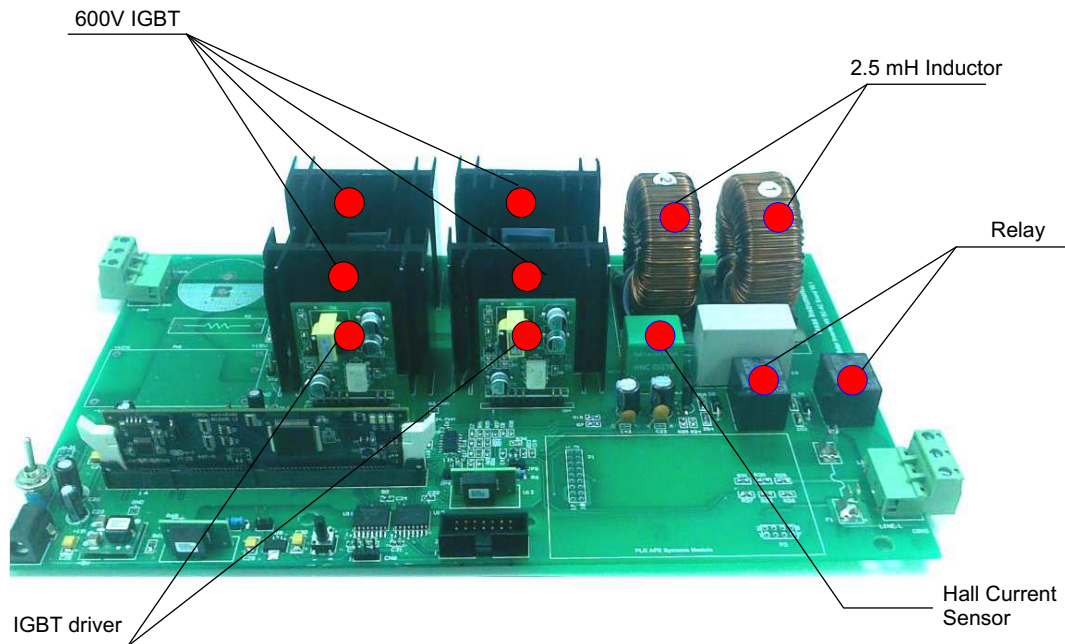


Figure 5. The Key Components on the Board

The PCB placement is shown in Figure 6.

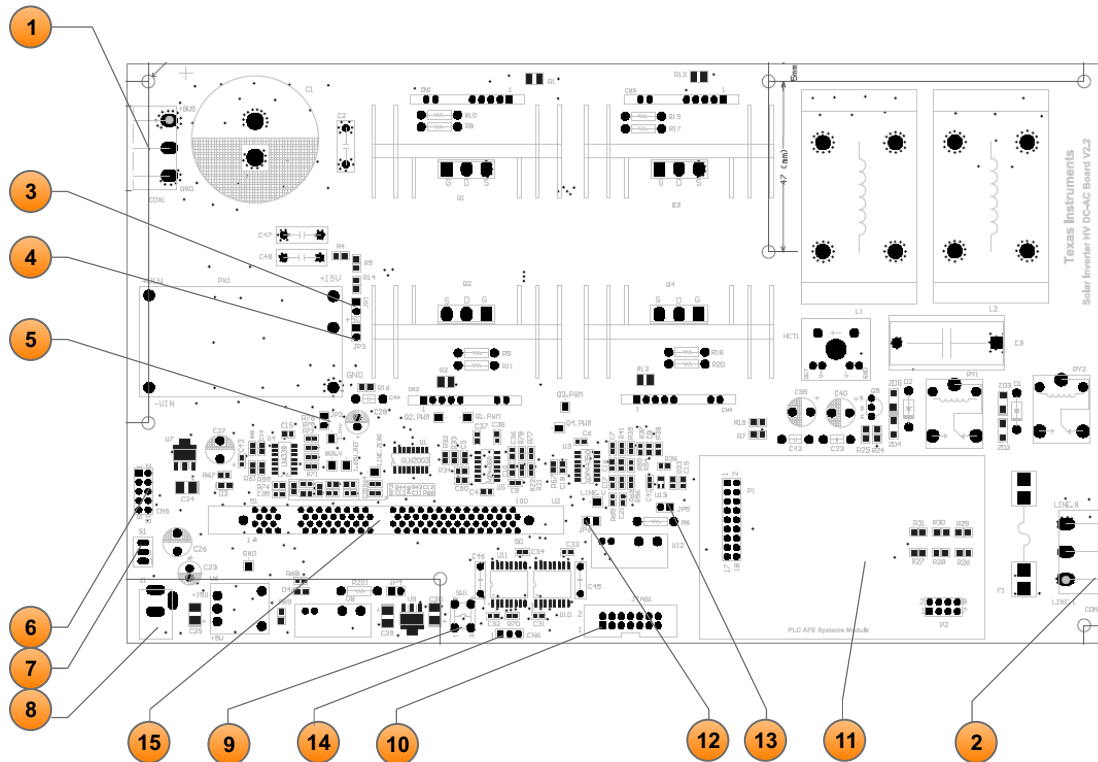


Figure 6. The PCB Placement

Table 1. The Key Points

Item No	Points Name	Comment
1	CON1	The DC bus connector for the DC-DC input
2	CON2	The utility connector L and N
3	JP1	Onboard +15 V jumper
4	JP3	Onboard +5 V jumper
5	JP2	IGBT driver +15 V jumper
6	CN5	DC-DC board signal interface
7	S1	External +15 V adapter switch
8	J1	External +15 V input jack
9	SW1	Operation button
10	JTAG1	JTAG interface for external emulator
11	PLC AFE Systems Module	Not used in this version
12	JP6	TRST jumper
13	JP5	-15 V power jumper
14	CN6	RS232 port
15	U2	The DIM100 28035 controlCard port

2.1.2 The Auxiliary Power Supply

The auxiliary power of the kit can be available by two ways. The one is using the external +15 V adapter. Insert the adapter to J1 (see [Table 1](#)), then switch S1 to power on. The other way is using the power module on the board (see the jump configuration) in [Table 2](#).

Table 2. The Jumper Setting for the Board

	External +15 V Adapter	Onboard +15 V
JP1	x	√
JP2	√	√
JP3	x	x
JP6	Unaffected	x

2.1.3 The Signal Sensing

Three key signals are used in the controller loop:

- DC BUS voltage
- Inductor current
- Grid voltage

The DC BUS voltage sensing is very simple. From the circuit, the sample ratio of the signal can be calculated as shown in [Equation 2](#):

$$K_{ratio_DCBUS} = \frac{R_6}{R_4 + R_5 + R_{14} + R_6} = \frac{10}{3010} = 0.003322 \quad (2)$$

For the inductor current sensing, there is a hall sensor whose sample ratio is 4/5. Besides, the differential circuit is used to get an appropriate ratio. The current sample ratio is calculated as shown in [Equation 3](#):

$$K_{ratio_current} = K_{hall} \frac{R_{41}}{R_{35} + R_{15}} = 0.15974 \quad (3)$$

For the utility voltage, just the differential circuit is used. The current is calculated as shown in [Equation 4](#):

$$K_{ratio_grid_voltage} = \frac{R_{59}}{R_{26} + R_{27} + R_{28} + R_{54}} = 0.003311 \quad (4)$$

NOTE: For the inductor current and grid voltage, there is a 1.65 V offset in the sample circuit. This need to be subtracted in the firmware.

2.1.4 Zero Crossing Detection

The zero crossing detection is used to detect the frequency. It is very convenient to detect the islanding conditions. The kit uses a comparator to get a falling edge in every positive zero crossing. Besides, a positive feedback of the comparator is used to get a sharp edge.

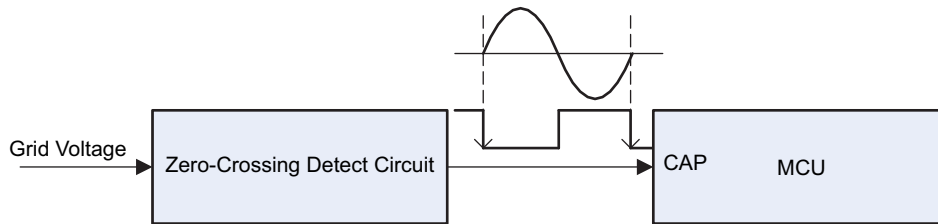


Figure 7. Zero Crossing

The CAP of the MCU captures the falling edge of the input signal and saves the capture value, which represents the positive zero crossing time of the grid voltage. In the firmware design, there is an interrupt for the capture event. The frequency can be calculated as shown in Equation 5:

$$grid_freq = \frac{f_{cpu_clk}}{(CAP_0 - CAP_1)} \tag{5}$$

The f_{grid_freq} is the grid frequency, the f_{cpu_clk} is the MCU CPU clock. The CAP0 is the capture value this time, the CAP1 is the capture value saved last time.

2.1.5 The IGBT Driver

The kit has four IGBT driver modules whose function is to isolate and amplify the driving capacity. The functional diagram for the driver is shown in Figure 8.

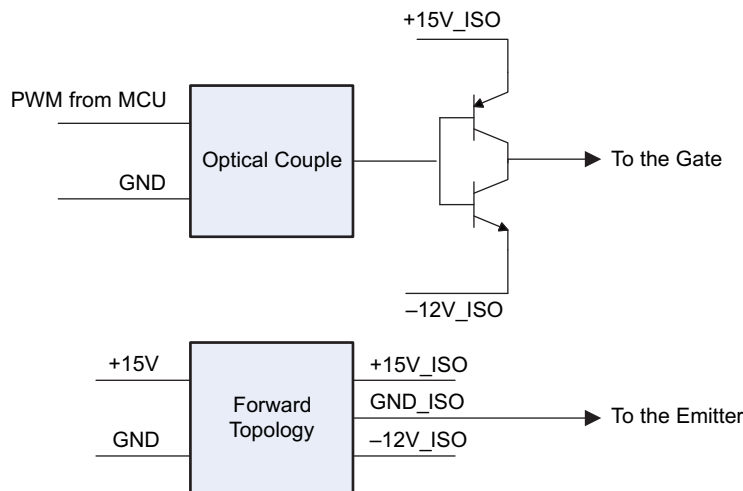


Figure 8. IGBT Driver Diagram

The driver can output +15 V for the turning on status and -12 V for the turning off status.

2.1.6 The Inductance

In order to smooth the current ripple, there is an inductor in the main circuit. The inductance is determined by the switching frequency f_s , the DC bus voltage bus V_{bus} , and the requirement of current ripple ΔI .

In a certain switching period, the inductor current can be described as shown in [Equation 6](#):

$$L \frac{\Delta I}{\Delta T} = V_L \quad (6)$$

Where the L is the inductance, and the

Assume in the single switching period, the rise of the current ΔI is equal to the fall of the current ΔI_f ,

$$\Delta I_r = \Delta I_f \quad (7)$$

Then you can get:

$$L \frac{\Delta I_r}{\Delta T} = L \frac{\Delta I}{dT_s} = V_{bus} - V_o \quad (8)$$

$$L \frac{\Delta I_f}{\Delta T_f} = L \frac{\Delta I}{(1-d)T_s} = -V_o \quad (9)$$

If you link [Equation 1](#), [Equation 4](#), and [Equation 5](#), you can get:

$$\Delta I = \frac{V_{bus}}{Lf_s} \left[-(d - 0.5)^2 + 0.25 \right] \quad (10)$$

From [Equation 10](#), the maximum ΔI occurs at the $d = 0.5$, and the maximum value is shown in [Equation 11](#):

$$\Delta I_{\max} = \frac{V_{bus}}{4Lf_s} \quad (11)$$

According to [Equation 11](#), you can calculate the inductance requirement of the full bridge inverter with the single polar modulation.

2.2 Firmware

2.2.1 The Firmware Structure

The typical front and background system is used in the firmware design. For the background, three different timer-based tasks are scheduled to deal with the non-urgent tasks. Besides, three interrupt service routines are used as the front to deal with the urgent things, such as the closed-loop controllers, the capture event, and the serial communications interface (SCI) receiving.

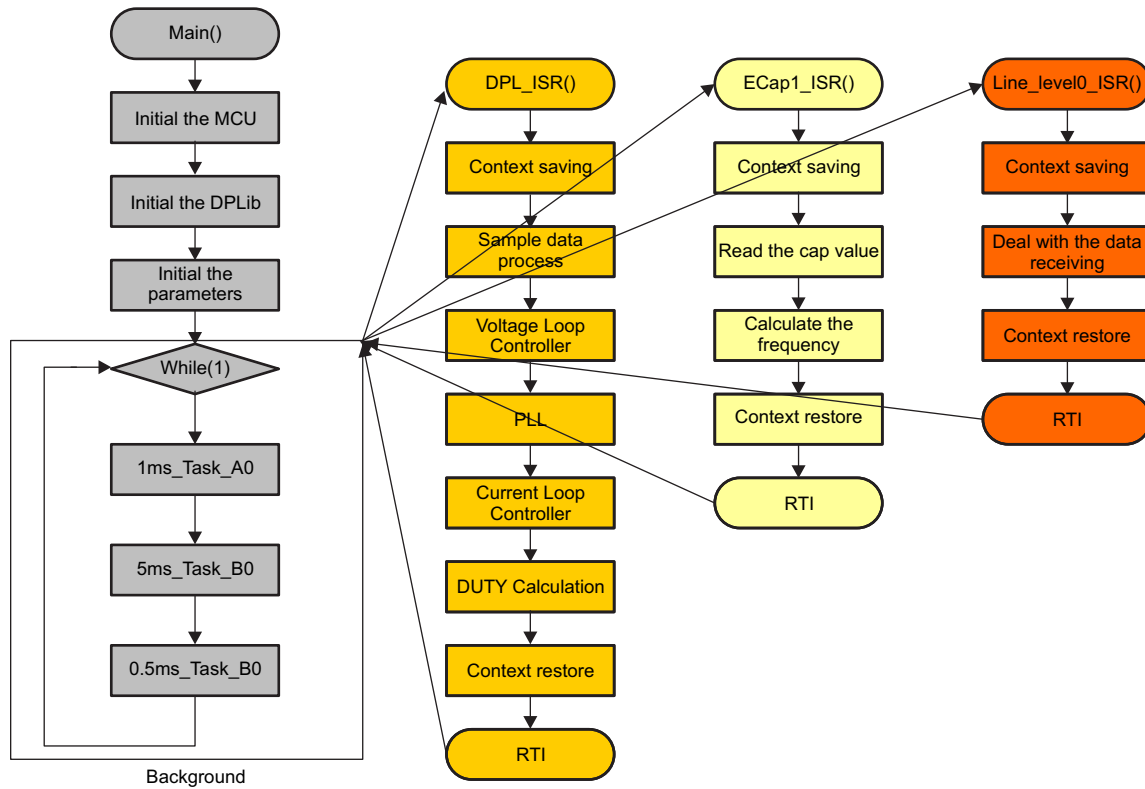


Figure 9. The Firmware Structure

2.2.2 The Status Machine

The status machine is used in order to distinguish the different status of the system. Different status represents different running modes, which according to the mode, the other tasks can take the appropriate action.

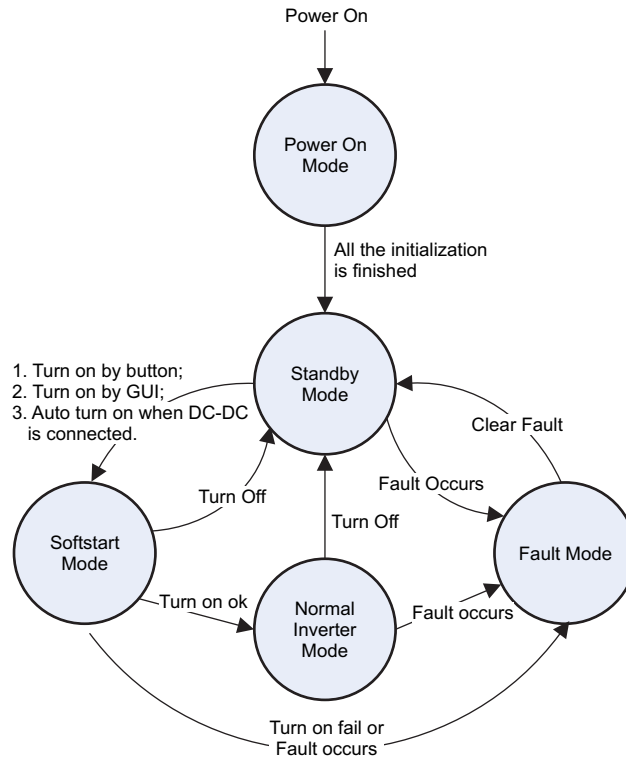


Figure 10. Status Machine

There are five different running modes in the firmware.

- **Power On Mode.** When the board powers up, it goes into the power on mode, then the MCU initializes itself. When the initialization is finished, the system transfers to standby mode automatically.
- **Standby Mode.** When the system is in standby mode, all the pulse-width modulation (PWM) and Relay are off. The system is waiting for the command to turn on; it will detect if the fault occurs.
- **Soft Start Mode.** When there is a turn on command, the system goes to the soft start mode first, then the PWM and relay are turned on. When the turning on is OK and no fault occurs, the system goes into the normal inverter mode automatically.
- **Normal Inverter Mode.** When the system is in normal inverter mode, it means the system feeds the energy out. If there is no fault or turn off command, the system stays in this mode.
- **Fault Mode.** When there is a fault, for example bus over voltage, the system transfers to the fault mode immediately. All the PWM is off, the output relay is cut off from the output. The fault can be cleared by the button or the graphical user interface (GUI). When the fault is cleared, it will return to standby mode.

2.2.3 The LED Flashing Design

The LED on the controlCard flashes in different ways according to the running mode defined in [Section 2.2.2](#). For more information, see [Table 3](#).

The LD2 is defined as the mode LED, and the LD3 is defined as the fault LED.

Table 3. The LED Flashing Definition

System Mode	LD2	LD3
Power On Mode	Always On	Always On
Standby Mode	Flashing in every 0.5s	Always Off
Standby Mode (with warning) ⁽¹⁾	Flashing in every 0.5s	Flashing in every 0.5s
Soft Start Mode	Flashing fast	Always Off
Normal Inverter Mode	Always On	Always Off
Fault Mode	Always Off	Always On ⁽²⁾

⁽¹⁾ When the LD3 is flashing, press the button on the board or click the 'turn on' button in the GUI to clear the warning. The system can be turned on only if there is no fault or warning. The warning can be generated by the following conditions: turning off, grid voltage out of range or the DC bus voltage abnormal. Check the firmware for the warning generation details. The flag named *FSuperFlag.BIT.FwWarning* represents the warning status.

⁽²⁾ If the LD3 is flashing or always on, power off and check the hardware.

NOTE: You can get the running mode by the LED flashing quickly.

2.2.4 The Tasks

In the background, three main tasks are used:

- Task_A0. The 1 millisecond task, it has four sub-tasks, only task A1 and task A3 are used in the system.

The sub task A1 deals with the status machine transition. The status is checked in every 20 ms. When the running mode is changed, the new running mode will take effect after 20 ms.

The sub task A3 deals with the onboard button detection and the LED flashing control.

- Task_B0. The 4 milliseconds task, it has four sub-tasks.

The sub task B1 deals with the fault detection, including the short circuit check, over current check, grid voltage and frequency check, as well as the DC bus voltage check. The sub task B2 deals with the measurement calculation, it calculates the grid voltage RMS and output current RMS, the active power, the DC bus voltage, as well as the zero crossing check.

The sub task B3 deals with the turning on check.

The sub task B4 deals with the GUI command processing and board-to-board communication.

- Task_C0. It is the 0.5 millisecond task. Only the C0 is used to check the SCI communication.

2.2.5 The Interrupt

Three interrupts are used to deal with the real-time events:

- The ADCINT1. The interrupt is generated by the ADC EOC. When the ADC sampling is finished, the interrupt is triggered. The ISR executes the controller algorithm.
- ECAP1_INT. The interrupt is generated by the capture event. When the zero crossing happens, the falling edge triggers the capture event.
- LIN0INTA. The interrupt is generated by the RXD event of the LINA, the LIN is used as the SCI port to communicate with the DC-DC board.

3 How to Build the Firmware

3.1 The File Structure of the Project

There are many files in the software project, including the c files, the assembly files, the head files as well as the cmd files.

Table 4. C Files in Project

C files Name	Description
ADC_SOC_Cnf.c	Initialize the ADC
SciCommsGui.c	Communication with GUI
SolarHv_DCAC-DevInit_F2803x.c	MCU device initialization
SolarHv_DCAC-CAP_Cnf.c	Cap initialization
SolarHv_DCAC-Lin.c	Communication with DC-DC board
SolarHv_DCAC-main.c	The background
SolarHv_DCAC-PWM_Cnf.c	ePWM initialization

Table 5. asm Files

asm files Name	Description
SolarHv_DCAC-CNTL_2P2Z.asm	The 2P2Z controller for the current
SolarHv_DCAC-ADCDRV_5CH.asm	ADC sample
SolarHv_DCAC-DLOG_4CH.asm	Get the real time data
SolarHv_DCAC-GEN_SIN_COS.asm	Generate the sine and cosine wave
SolarHv_DCAC-INV_ICMD.asm	Calculate the current loop reference
SolarHv_DCAC-ISR.asm	The ADC interrupt ISR for the controller
SolarHv_DCAC-PWMDRV.asm	Calculate the CMPR and update the duty

Table 6. Other Files

Other files Name	Description
SolarHv_DCAC-Settings.h	The project build setting
SolarHv_DCAC-f28035_FLASH.CMD	Cmd file for code running in Flash
SolarHv_DCAC--f28035_RAM.CMD	Cmd file for code running in RAM

3.2 The Blocks Introduction

There are some blocks that are used to realize the specified function. You can use the blocks in your own projects.

3.2.1 The ADCDRV_5CH m n p q s

The block named ADCDRV_5CH is the ADC sampling driver module, which can be used to get five sample channels.

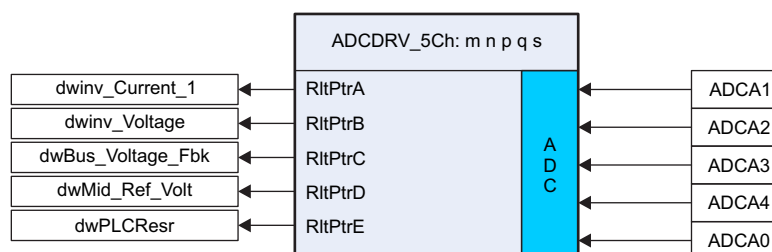


Figure 11. ADCDRV_5CH Block

There are five channels ADC are used:

- **ADCA1** is assigned for the inductor current sensing, the *dwInV_Current_1* is named for this channel in the software, the format of *dwInV_Current_1* is *Q24*.
- **ADCA2** is assigned for the grid voltage sensing, the *dwInV_Voltage* is named for this channel in the software, the format of *dwInV_Voltage* is *Q24*.
- **ADCA3** is assigned for the DC BUS voltage sensing, the *dwBus_Voltage_Fbk* is named for this channel in the software, the format of *dwBus_Voltage_Fbk* is *Q24*.
- **ADCA3** is assigned for the 1.65 V reference sensing, the *dwMid_Ref_Volt* is named for this channel in the software, the format of *dwMid_Ref_Volt* is *Q24*.
- **ADCA0** is reserved for the PLC application in the future.

3.2.2 The GEN_SIN_COS: n

The GEN_SIN_COS: n is used to generate the sine wave and cosine wave.

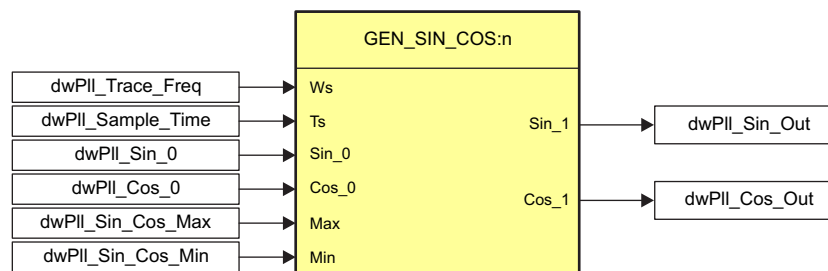


Figure 12. The GEN_SIN_COS: n

The *Ws* is the frequency input of the generator. The *dwPll_Trace_Freq* is assigned for this input, the format is *Q20*. For example:

$dwPll_Trace_Freq = _IQ20(376.9911)$ represents the 60Hz

The *Ts* is the sample frequency of the generator. The *dwPll_Sample_Time* is assigned for this input, the format is *Q24*. For example:

$dwPll_Sample_Time = _IQ(0.000052)$ represents the 52e-6 seconds

The *Sin_0* is the initial value of the sine value. The *dwPll_Sin_0* is assigned for this input, the format is *Q22*. The default value of the *dwPll_Sin_0* is 0.

The *Cos_0* is the initial value of the sine value. The *dwPll_Cos_0* is assigned for this input, the format is *Q22*. The default value of the *dwPll_Cos_0* is $_IQ22(0.99)$.

The *Max* is the maximum value of the output value. The *dwPll_Sin_Cos_Max* is assigned for this input, the format is *Q22*. The default value of the *dwPll_Sin_0* is $_IQ22(0.99)$.

The *Min* is the minimum value of the output value. The *dwPll_Sin_Cos_Min* is assigned for this input, the format is *Q22*. The default value of the *dwPll_Sin_Cos_Min* is 0.

3.2.3 INV_ICMD:n

The INV_ICMD:n is used to calculate the current reference.

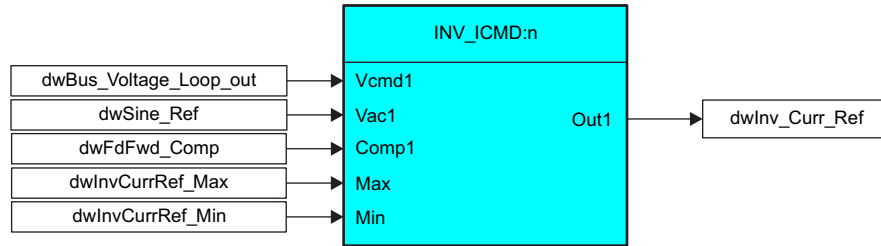


Figure 13. INV_ICMD:n

The *Vcmd1* is the amplitude of the reference current that is usually the voltage loop controller output. The *dwBus_Voltage_Loop_Out* is assigned as the interface. The format is Q24.

The *Vac1* is the unit sine wave that represents the reference angle of the current, which is usually the sine generator's output. The *dwSine_Ref* is assigned as the interface. The format is Q24.

The *Comp1* is the compensation for the change of the grid voltage. The default value is 1. The Max, Min is the limitation of the output.

Out1 is the output of the block, the *dwInv_Curr_Ref* is assigned as the interface. The format is Q24.

3.2.4 PWMDRV:n

The PWMDRV: n is used to calculate the CMPR according to the controller's output. It will update the CMPR register when it finishes the calculation.

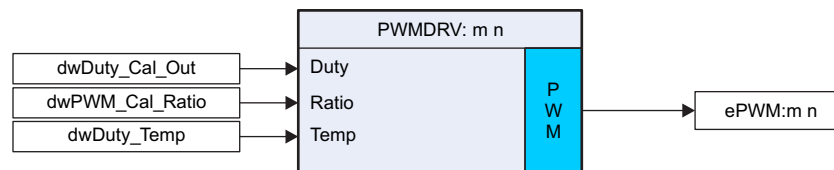


Figure 14. PWMDRV:n

The *Duty* is the output of the controller, which is usually the current loop controller output. The *dwDuty_Cal_out* is assigned for this input, the format is Q24.

The *Ratio* is the conversion ratio between the duty and the CMPR value. The format is Q8. The ratio can be calculated by the following method:

$$\text{Ratio} = \text{Period} * 1000 / V_{dc}$$

The *Temp* is reserved for debug.

3.2.5 CNTL_2P2Z:n

This is the same to the blocks defined in the Digital Power Library.

3.2.6 DLOG_4CH:n

This is similar to the blocks defined in the Digital Power Library, but the start of the log is different. In this project, the block starts the data log when the variable *wDataEnable* is 1.

3.3 The Build Step

The following section discusses the incremental build steps and provides step-by-step functions.

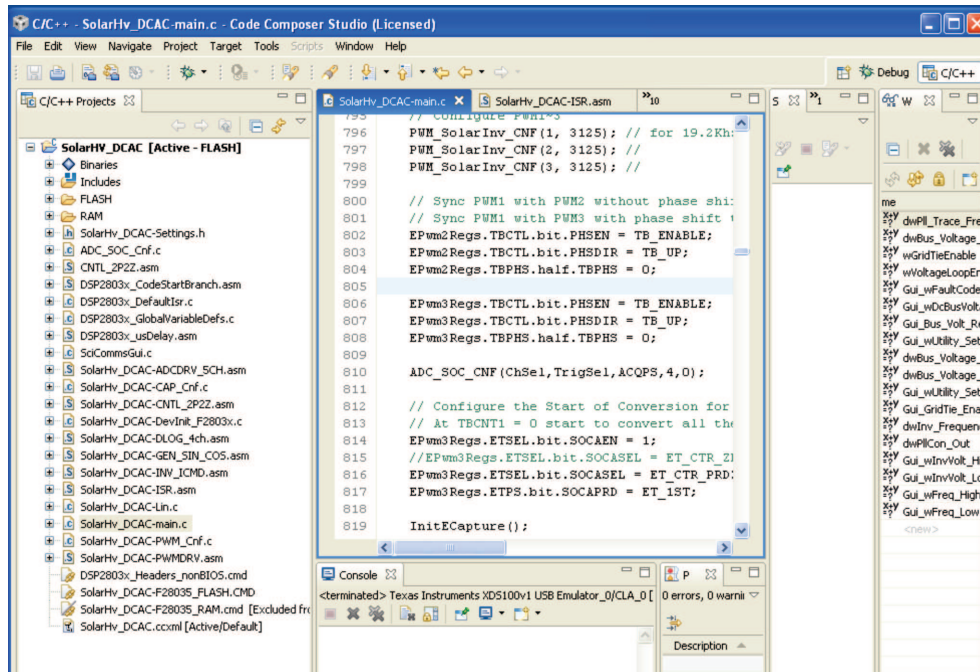
The build step can be set by the pre-defined macro INCR_BUILD in the head file named *SolarHv_DCAC-Settings.h*, see the setting in the [Table 7](#).

Table 7. Incremental Build Option

INCR_BUILD == 1	Open loop build
INCR_BUILD == 2	Close loop without PLL
INCR_BUILD == 3	Close loop with PLL

3.3.1 Start the Code Composer Studio Project

1. Connect the USB cable to the ISO PiccoloB controlCard. Shorten jumper JP2, JP4, JP5, JP6; open jumper JP3,JP1.
2. Insert the +15 V adapter to J1, then switch S1 to power on the auxiliary power.
3. Start CCSV4 and create a new workspace.
4. Click the menu: Project → Import Existing CCS → CCE Eclipse Project. Under the Select root directory, navigate to and select ..\controlSUITE\development_kits\Solar HV Kit\DC-AC board when the integrated development environment (IDE) opens. The following workspace is shown when the project opens.



5. Change the incremental build option by setting a value to INCR_BUILD as shown in Figure 15.

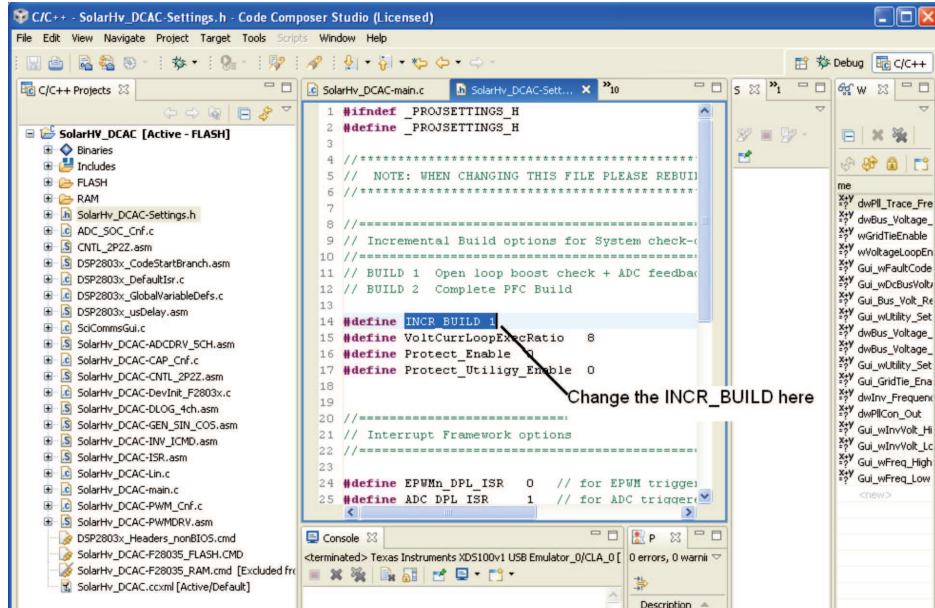


Figure 15. Change the Incremental Build

6. Set the build configuration by clicking the menu: Project → Active Build Configuration. If you want to run the code in RAM, choose the RAM or FLASH option.
7. Rebuild the project by clicking the menu: Project → Rebuild All. If there is no error, the new .out file will be created.
8. In the .ccxml file that opens, select Connection as the “Texas Instruments XDS100v2 USB Emulator”. Under the device, scroll down and select “TMS320F28035”. Click Save.
9. Start the TI debugger by clicking Target → Debug Active Project.
10. Figure 16 appears when the code is loaded successfully.

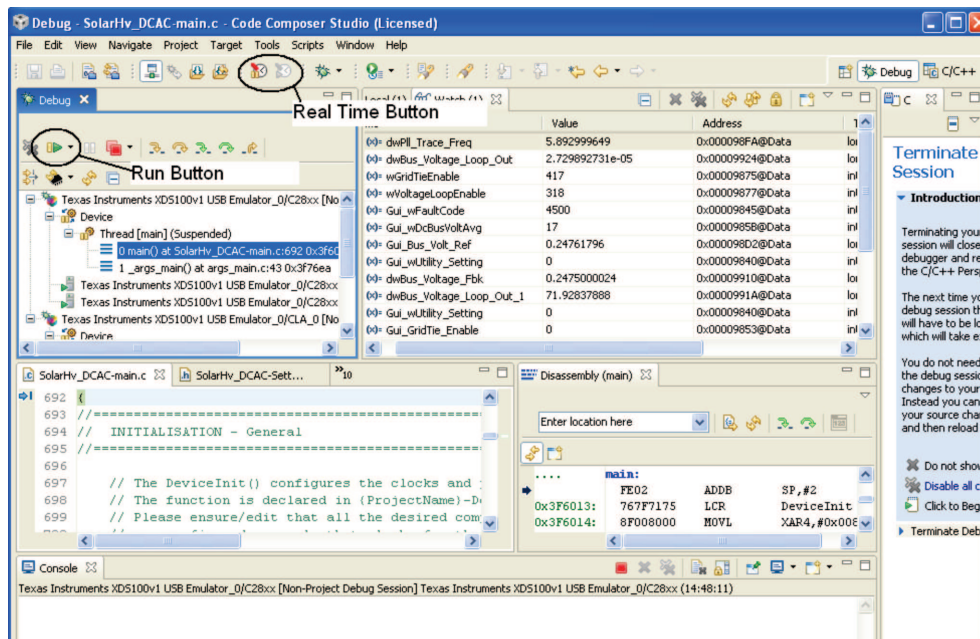


Figure 16. Debug Active Project

11. Use the real-time debug option by clicking the button in the tool bar.

12. Run the code by clicking the “Run” button in the tool bar.

3.3.2 Open Loop Build

The first step is the open loop build; let the board output a sine wave. In this step, the GEN_SIN_COS and the PWMDRV block are used to generate the SPWM. DLOG_4CH and ADCDRV_5CH are also used. You can check the sample data in real time or via the GUI. (If the GUI is used, you must run the code in Flash).

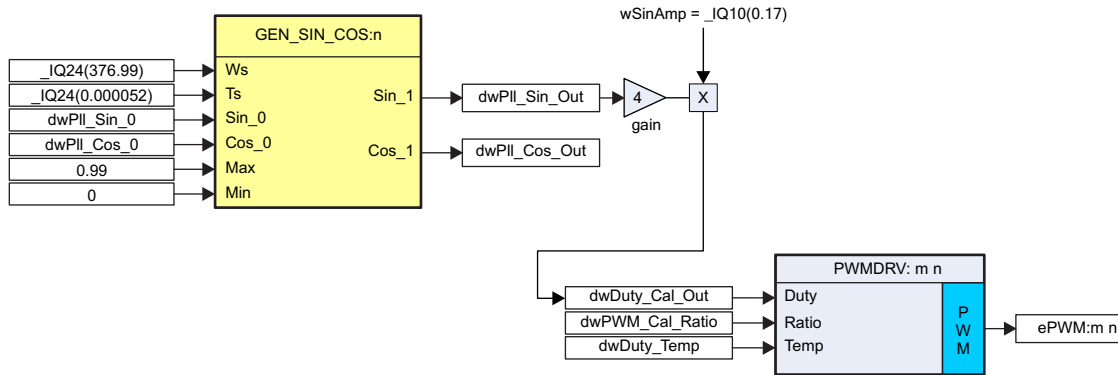


Figure 17. Open Loop Build

The open loop build can be available when you set the *INCR_BUILD = 1* in the *SolarHv_DCAC-Settings.h* file.

When the code is running, set the DC source input to about 400 V, then press the SW1 to turn the board on.

3.3.3 Close Loop Build Without PLL

When the grid is not connected to the board, the board can run the close loop without the PLL. It outputs a constant current to the load.

Before you build this step, make sure you do the open loop test successfully; it must connect a resistor load to the output. The suggested resistor load is 25 Ω/1000W.

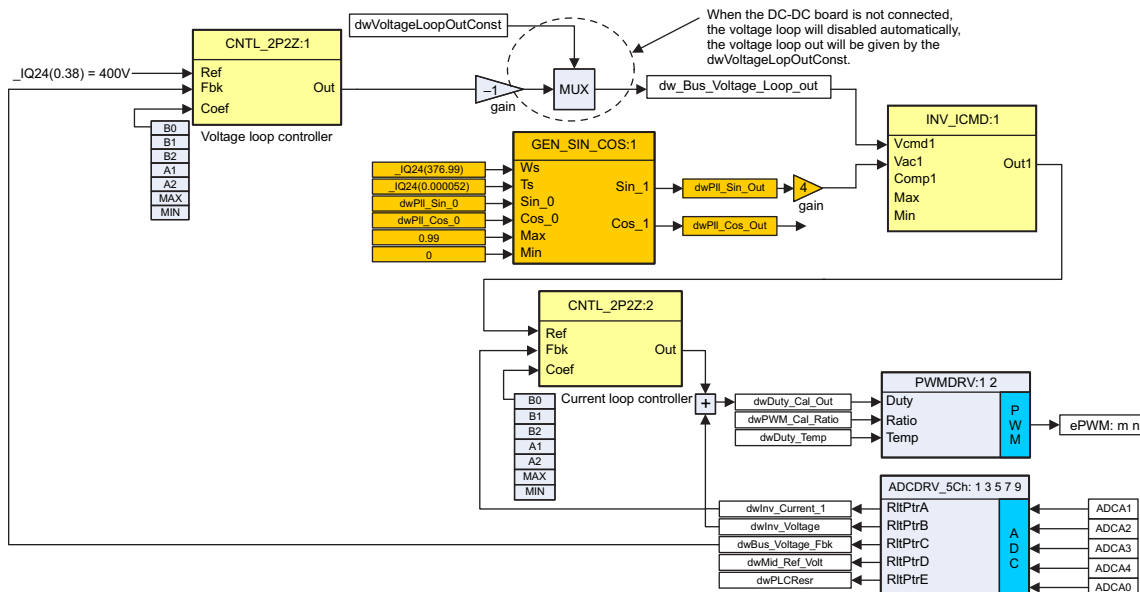


Figure 18. Close Loop Build Without PLL

The close loop without PLL build can be available when you set the *INCR_BUILD = 2* in the *SolarHv_DCAC-Settings.h* file.

Note that when the DC-DC board is not connected, the voltage loop will be automatically disabled. The *dw_Bus_Voltage_Loop_Out* will be given by the *dwVoltageLoopOutConst* directly. You can modify the *dwVoltageLoopOutConst* in real time to get the different output current value.

3.3.4 Close Loop Build With PLL

If the build mentioned in Section 3.3.3 is finished, you can do the final build step for the grid tie test. You must connect the test tool to the board as shown in Figure 19.

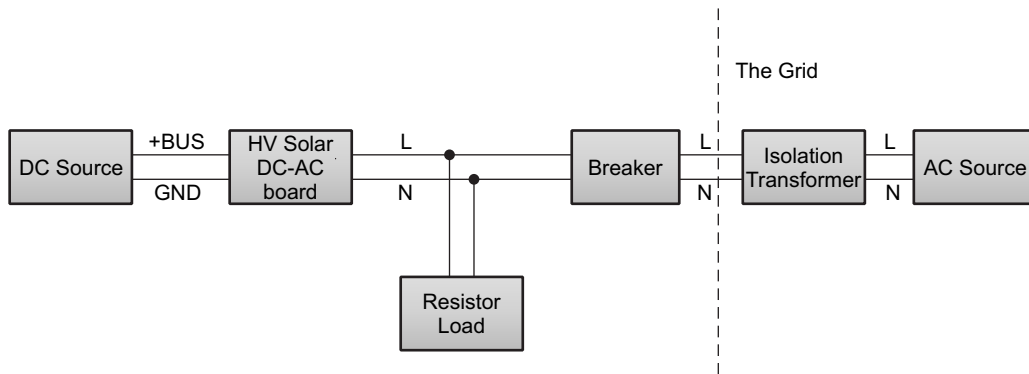


Figure 19. The Connection of the Test

For safety, TI strongly suggests that you use a breaker between the grid and the inverter output.

NOTE: All the tests should be done in a lab and you must use the AC source to emulate the grid. Security cannot be ensured when you use this board to connect to the grid.

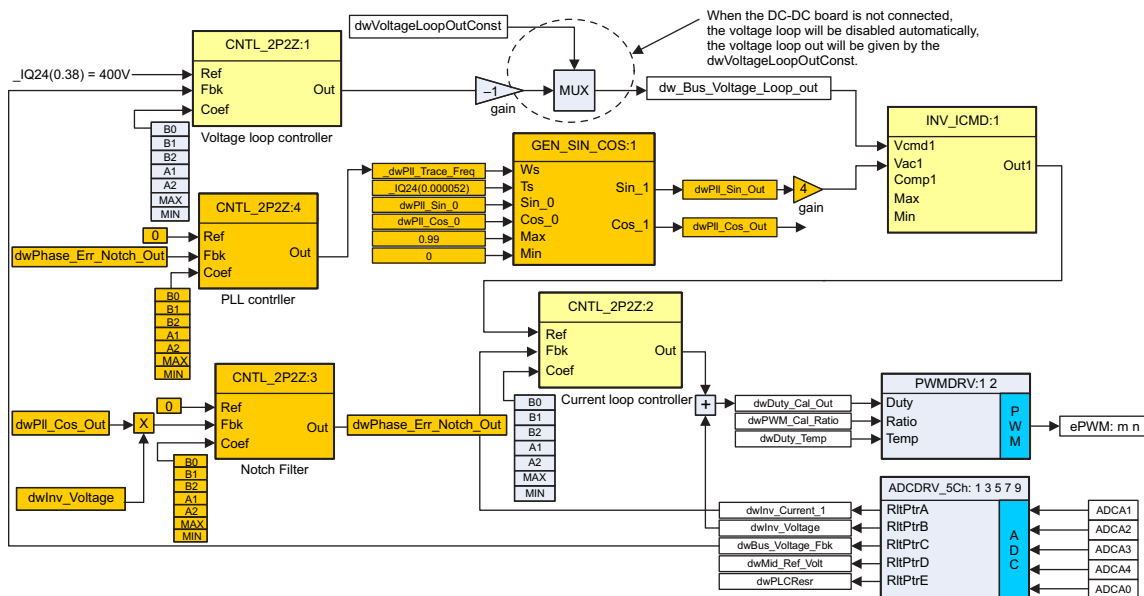


Figure 20. Close Loop With PLL Build

The close loop with PLL build can be available when you set the *INCR_BUILD = 3* in the *SolarHv_DCAC-Settings.h* file.

Note that when the DC-DC board is not connected, the voltage loop will be automatically disabled. The *dw_Bus_Voltage_Loop_Out* will be given by the *dwVoltageLoopOutConst* directly. You can modify the *dwVoltageLoopOutConst* in real time to get the different output current value.

4 Test Result

4.1 Specification

The system main spec is below:

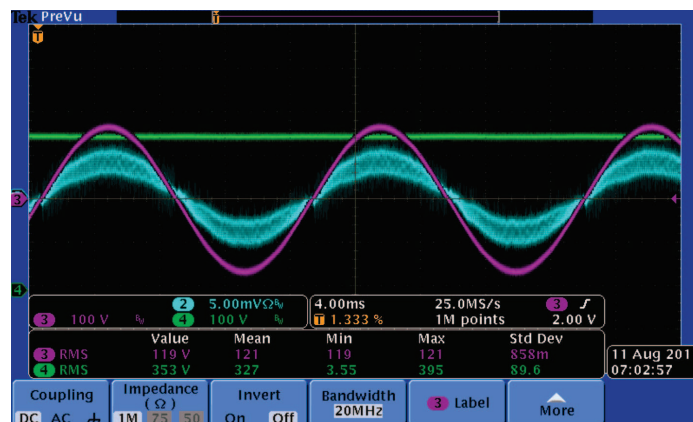
- Power Rating: 600W
- Norminal Grid Voltage: 120 V/60Hz(RMS), 220 V/50Hz
- Output Power Factor: 1
- THDi: <5%
- Panel Input Voltage Range: 400 V
- Grid Tie
- Anti-Islanding Protection

Test condition:

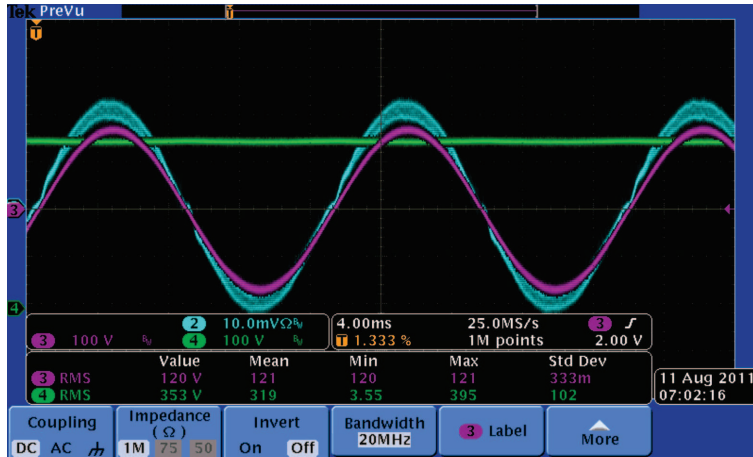
- AC Source Connected, with 120VAC/60Hz
- DC Bus Voltage: 400 V
- Power Range: 100-600W output
- Grid Tie
- Room Temperature

4.2 DC-AC Board Current Loop Grid-Tie Test Result

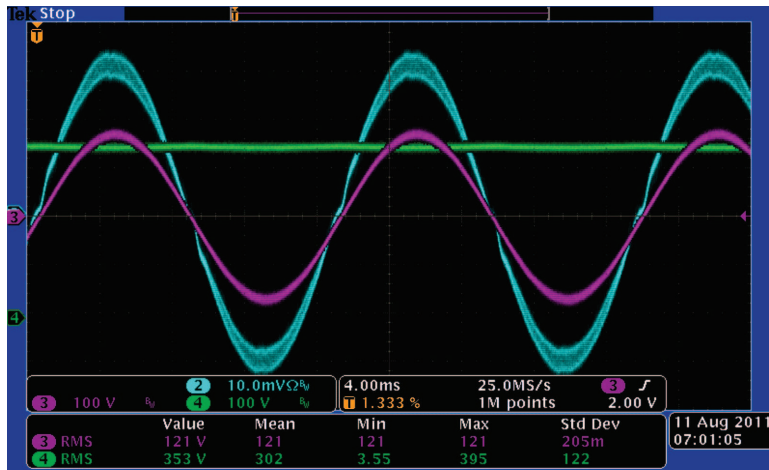
- Light load current and grid voltage waveform
 - CH2: Output Current (Blue)
 - CH3: Grid Voltage (Red)
 - CH4: Bus voltage



- Middle load current and grid voltage waveform



- Full load current and grid voltage waveform



4.3 Output Power Factor and THDi

Table 8. The PF and THDi

Inv V_out	Inv P_out	Output PF	THDi
119.5	100.3	0.983	12.60%
119.8	151.6	0.992	8.70%
119.2	198.4	0.995	6.80%
119.5	248.1	0.996	5.80%
119.8	297.7	0.997	5%
120.1	344.1	0.997	4.30%
119.6	391.7	0.997	3.90%
119.9	439.2	0.997	3.60%
120	464.2	0.997	3.40%

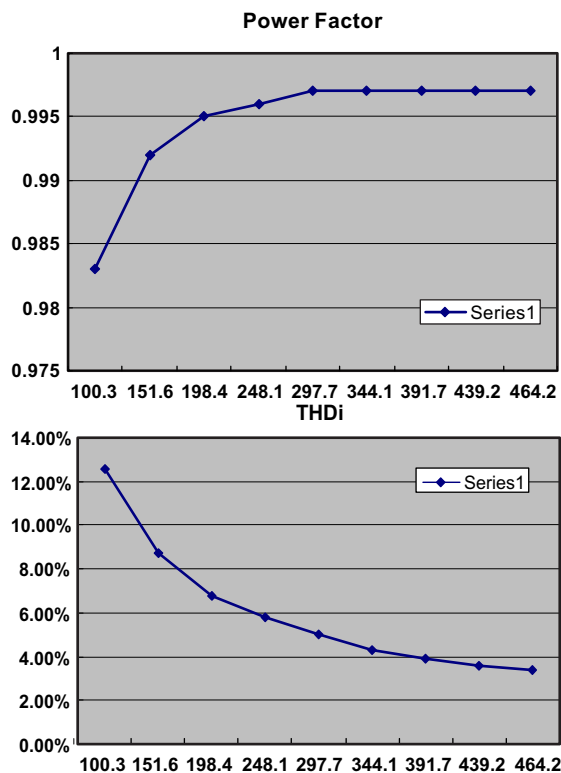


Figure 21. PF and THDi

4.4 Efficiency

Table 9. Efficiency

Item	DC_in(V)	AC_out(V)	Output(W1)	Input(W)	Efficiency(%)
1	400	120	609	632	96.3
2	400	120	536	557	96.2
3	400	120	500	521	95.9
4	400	120	446	467	95.5
5	400	120	356	376	94.5
6	400	120	302	321	94.1

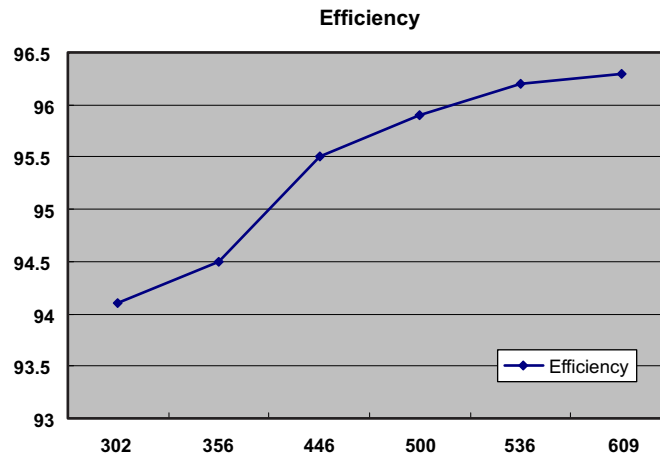


Figure 22. The Efficiency

4.5 The HV Solar System Test

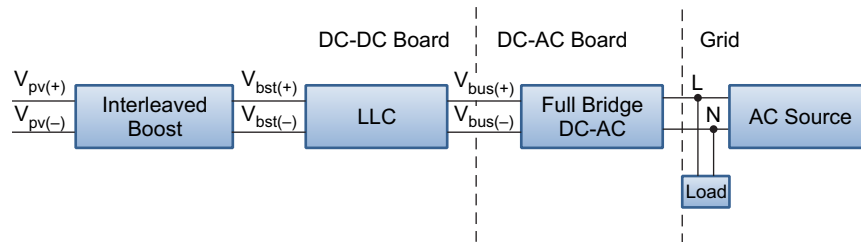


Figure 23. The System Structure and the Connection

- 120VAC/60Hz, turning on:
 - CH2: Output Current (Blue)
 - CH3: Grid Voltage (Red)
 - CH4: Bus voltage

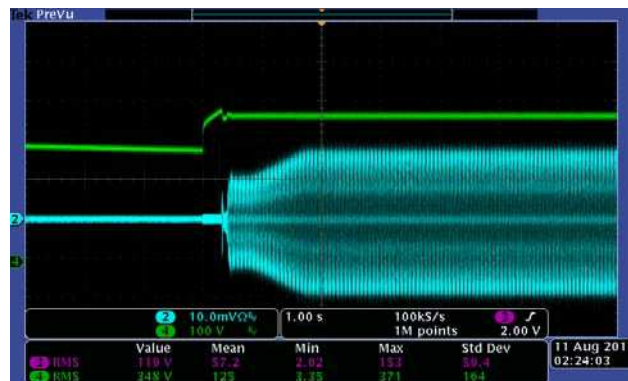


Figure 24. The Turn on Overview

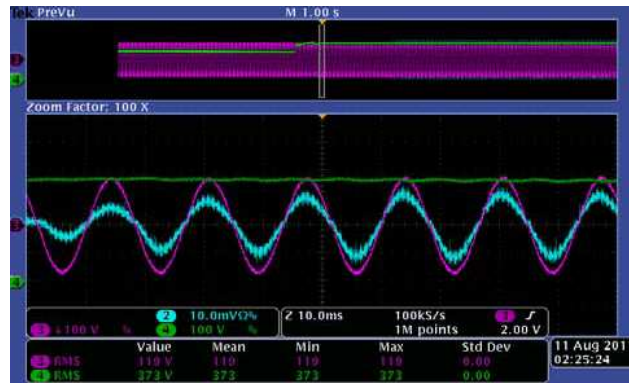
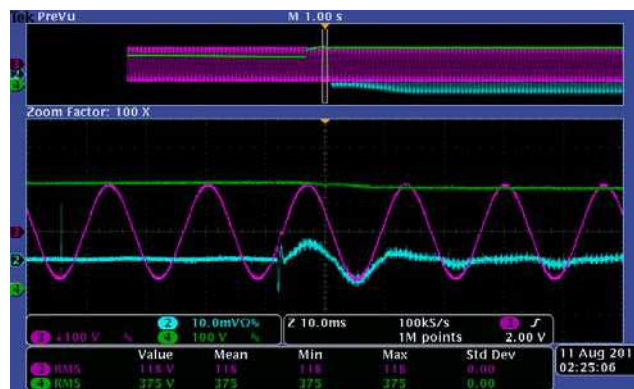
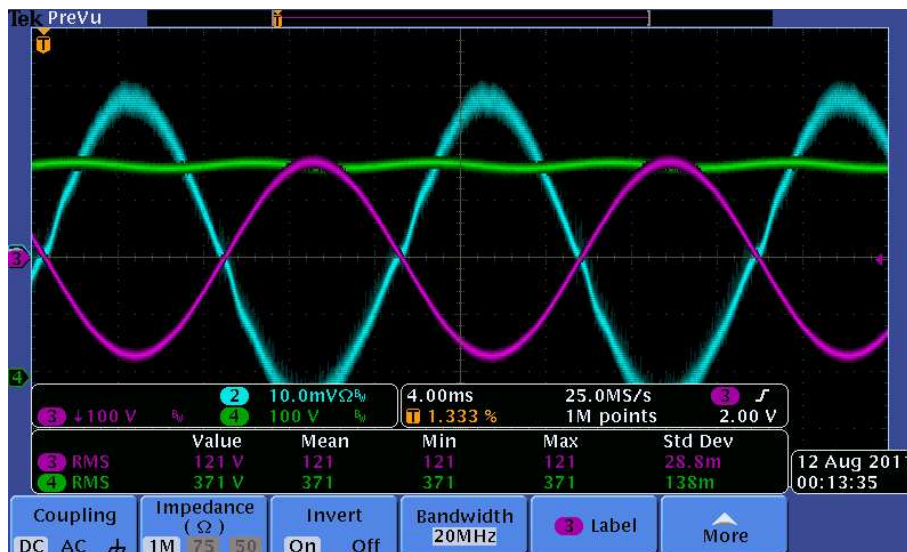
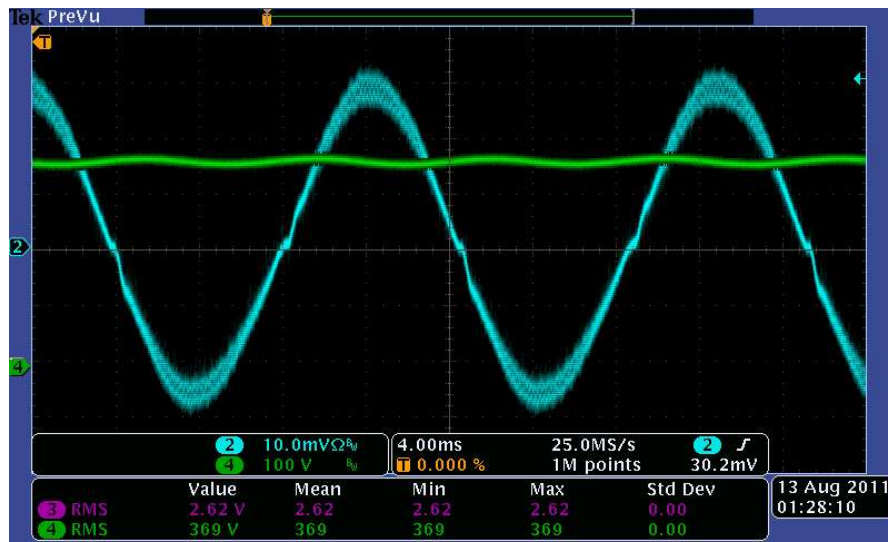


Figure 25. The DC-AC Turn on the PWM



- 120VAC/60Hz, 500W
 - CH2: Output Current (Blue)
 - CH3: Grid Voltage (Red)
 - CH4: Bus voltage





IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as "components") are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI's terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers' products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers' products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have **not** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products

Audio	www.ti.com/audio
Amplifiers	amplifier.ti.com
Data Converters	dataconverter.ti.com
DLP® Products	www.dlp.com
DSP	dsp.ti.com
Clocks and Timers	www.ti.com/clocks
Interface	interface.ti.com
Logic	logic.ti.com
Power Mgmt	power.ti.com
Microcontrollers	microcontroller.ti.com
RFID	www.ti-rfid.com
OMAP Applications Processors	www.ti.com/omap
Wireless Connectivity	www.ti.com/wirelessconnectivity

Applications

Automotive and Transportation	www.ti.com/automotive
Communications and Telecom	www.ti.com/communications
Computers and Peripherals	www.ti.com/computers
Consumer Electronics	www.ti.com/consumer-apps
Energy and Lighting	www.ti.com/energy
Industrial	www.ti.com/industrial
Medical	www.ti.com/medical
Security	www.ti.com/security
Space, Avionics and Defense	www.ti.com/space-avionics-defense
Video and Imaging	www.ti.com/video

TI E2E Community

e2e.ti.com