Application Note

Bidirectional DC-DC Converter Power Stage Implementation of the TIDM-02009 Reference Design Using AM263x MCU

Shamik Basak

ABSTRACT

Ever since the introduction of the topology, phase-shifted full bridge rectifiers (PSFB) have gained a wide range of acceptance in industrial applications like power supplies, telecom rectifiers, battery charging systems, renewable energy systems, and so forth. Due to the ease of controllability, the PSFB rectifier is often selected as the go-to method for transferring bulk amounts of power from the high-voltage (HV) DC to the low-voltage (LV) DC side with electrical isolation. One such simple control scheme that can be used in this application is called peak current mode control (PCMC). The power transfer happens from the HV to LV side when the turn ON period of diagonally opposite switches of two legs overlap with each other. In PCMC, this overlapping period is controlled by making the appropriate peak current value setting. On top of this, an outer voltage control loop is implemented to maintain the output voltage at the desired level. In this work, the control mechanism is implemented using ARM R5F core-based SITARA MCU of Texas Instruments (TI) to showcase the real-time control capabilities. In addition to that, a synchronous rectification feature is implemented on the LV side to harness the optimized power output from the converter.

Table of Contents

List of Figures

List of Tables

Trademarks

controlCARD™, C2000™, and Code Composer Studio™, and E2E™ are trademarks of Texas Instruments.
All trademarks are the property of their respective owners.

1 General Texas Instruments High Voltage Evaluation (TI HV EVM) User Safety Guidelines

Always follow TI's setup and application instructions, including use of all interface components within their recommended electrical rated voltage and power limits. Always use electrical safety precautions to help ensure your personal safety and those working around you. For more information, contact TI's Product Information Center https://support/ti./com.

Save all warnings and instructions for future reference.

> **WARNING**
>
> Failure to follow warnings and instructions may result in personal injury, property damage or death due to electrical shock and burn hazards.

The term TI HV EVM refers to an electronic device typically provided as an open framed, unenclosed printed circuit board assembly. It is *intended strictly for use in development laboratory environments, solely for qualified professional users having training, expertise and knowledge of electrical safety risks in development and application of high voltage electrical circuits. Any other use and/or application are strictly prohibited by Texas Instruments.* If you are not suitable qualified, you should immediately stop from further use of the HV EVM.

1. Work Area Safety
   a. Keep work area clean and orderly.
   b. Qualified observer(s) must be present anytime circuits are energized.
   c. Effective barriers and signage must be present in the area where the TI HV EVM and its interface electronics are energized, indicating operation of accessible high voltages may be present, for the purpose of protecting inadvertent access.
   d. All interface circuits, power supplies, evaluation modules, instruments, meters, scopes and other related apparatus used in a development environment exceeding 50Vrms/75VDC must be electrically located within a protected Emergency Power Off EPO protected power strip.
   e. Use stable and nonconductive work surface.
   f. Use adequately insulated clamps and wires to attach measurement probes and instruments. No freehand testing whenever possible.
2. Electrical Safety
   As a precautionary measure, it is always a good engineering practice to assume that the entire EVM may have fully accessible and active high voltages.
   a. De-energize the TI HV EVM and all its inputs, outputs and electrical loads before performing any electrical or other diagnostic measurements. Revalidate that TI HV EVM power has been safely de-energized.
   b. With the EVM confirmed de-energized, proceed with required electrical circuit configurations, wiring, measurement equipment connection, and other application needs, while still assuming the EVM circuit and measuring instruments are electrically live.
   c. After EVM readiness is complete, energize the EVM as intended.

   > **WARNING**
   >
   > While the EVM is energized, never touch the EVM or its electrical circuits, as they could be at high voltages capable of causing electrical shock hazard.
3. Personal Safety
   a. Wear personal protective equipment (for example, latex gloves or safety glasses with side shields) or protect EVM in an adequate lucent plastic box with interlocks to protect from accidental touch.

Limitation for safe use:
EVMs are not to be used as all or part of a production unit.

2 Introduction

This document presents how an AM263x MCU can be used for controlling the bidirectional DC-DC converter power stage of the *ASIL D Safety Concept-Assessed High-Speed Traction, Bidirectional DC/DC Conversion Reference Design*. This hardware is compatible with MCUs in an HSEC controlCARD™ format. While originally designed for the C2000™ MCU product family, this reference design can accept the AM263x controlCARD (TMDSCNCD263) also with minimal modifications. This reference design is useful for implementing a digitally peak current mode controlled (PCMC) phase shifted full bridge (PSFB) DC-DC converter which converts a 400V DC input to a regulated 12V DC output and vice-versa. Although the design is capable of performing bi-directional power conversion, the scope of this application note is limited to illustrate only high voltage to low voltage conversion, that is, step down operation. Novel PCMC waveform generation based on type-5 PWM and internal slope compensation, and simple PCMC implementation are the highlights of this design.

3 System Description

Phase-shifted full bridge (PSFB) DC-DC converters are used frequently to step down high DC bus voltages and provide isolation in medium- to high-power applications like server power supplies, telecom rectifiers, battery charging systems, and renewable energy systems. Traditionally, microcontrollers have been restricted to only performing supervisory or communications tasks in these systems. With the availability of high-performing microcontroller devices, it is now possible to use microcontrollers for closing control loops in these systems, in addition to handling the traditional microcontroller functions. The transition to digital power control means that functions that were previously implemented in hardware are now implemented in software. In addition to the flexibility this adds to the system, this simplifies the system considerably. These systems can implement advanced control strategies to optimally control the power stage under different conditions and also provide system-level intelligence.

A PSFB converter consists of four power electronic switches (like MOSFETs or IGBTs) that form a full-bridge on the primary side of the isolation transformer and diode rectifiers, or MOSFET switches for synchronous rectification (SR) on the secondary side. This topology allows all the switching devices to switch with zero voltage switching (ZVS), resulting in lower switching losses and an efficient converter. In this reference design, ZVS for switches in the one leg of the full bridge, and zero or low-voltage switching for switches in the other leg, is achieved.

For such an isolated topology, signal rectification is required on the secondary side. For systems with low output voltage or high output current ratings, implementing synchronous rectification instead of diode rectification achieves the best possible performance by avoiding diode rectification losses. In this design, current doubler synchronous rectification is implemented on the secondary side with different switching schemes to achieve optimum performance under varying load conditions.

A DC-DC converter system can be controlled in various modes, like voltage mode control (VMC), average current mode control (ACMC), or peak current mode control (PCMC). PCMC is a highly desired control scheme for power converters because of the inherent voltage feed forward, automatic cycle-by-cycle current limiting, flux balancing, and other advantages.

3.1 Key System Specifications

Table 3-1. Key System Specifications

Parameter	Specification
Input Conditions	
Input voltage	200V DC - 400V DC
Input current	0.1A - 8.75A
Output Conditions	
Output voltage	12V
Output current	292A

Table 3-1. Key System Specifications (continued)

Parameter	Specification
Power rating	3.5kW (Tested upto 250W in this version)
System Characteristics	
Efficiency	To be calculated while testing at full capacity
PWM switching frequency	100kHz

4 System Overview

4.1 Block Diagram

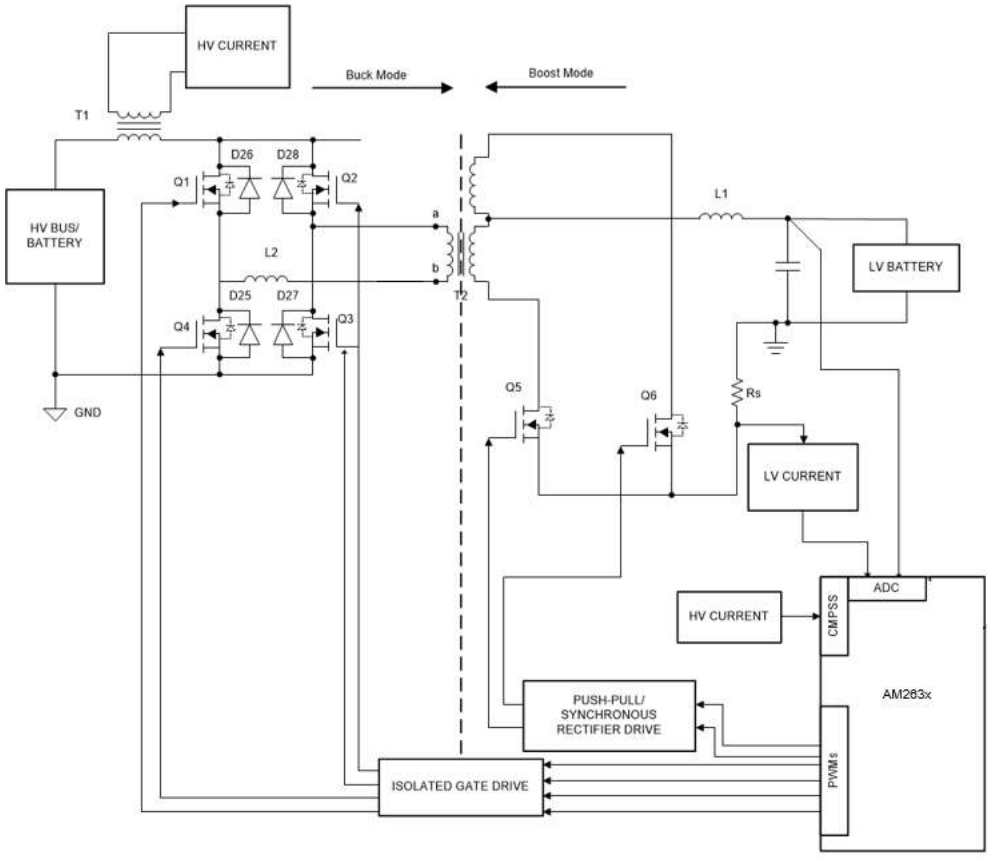


Figure 4-1. Block Diagram of the System

4.2 Basic Operation

A PSFB converter consists of four-power electronic switches (like MOSFETs or IGBTs) that form a full-bridge on the primary side of the isolation transformer and diode rectifiers or MOSFET switches for synchronous rectification (SR) on the secondary side. This topology allows the switching devices to switch with zero voltage switching (ZVS) resulting in lower switching losses and an efficient converter.

For such an isolated topology, signal rectification is required on the secondary side. For systems with low output voltage and/or high output current ratings, implementing synchronous rectification instead of diode rectification achieves the best possible performance by avoiding diode rectification losses. In this work, synchronous rectification is implemented on the secondary side.

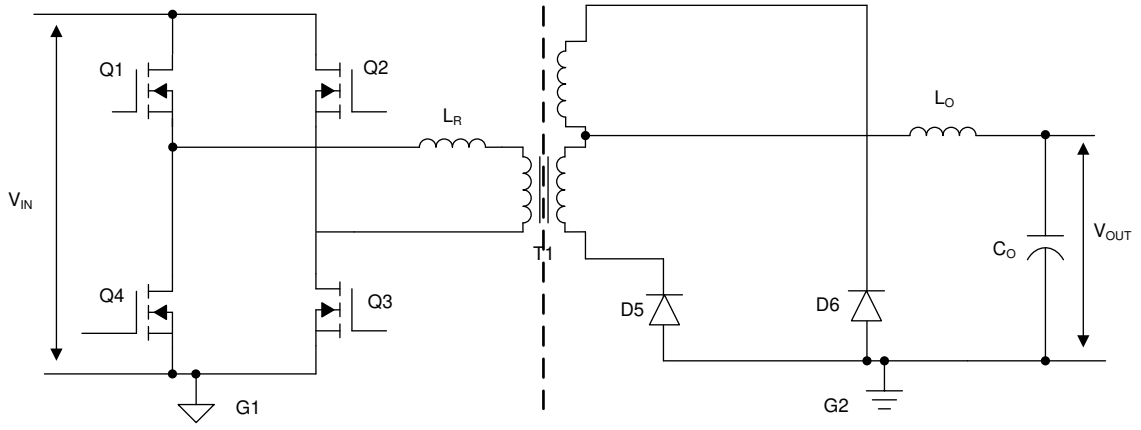


Figure 4-2. A Phase-Shifted Full Bridge Circuit

Figure 4-2 shows a simplified circuit of a phase shifted full-bridge. MOSFET switches Q_1, Q_4, Q_2 and Q_3 form the full-bridge on the primary side of the transformer T1. Q_1 and Q_4 are switched at 50 % duty and 180° out of phase with each other. Similarly, Q_2 and Q_3 are switched at 50 % duty and 180° out of phase with each other. The PWM switching signals for leg Q_2 – Q_3 of the full-bridge are phase shifted with respect to those for leg Q_1 - Q_4. Amount of this phase shift decides the amount of overlap between diagonal switches, which in turn decides the amount of energy transferred. D_5, D_6 provide diode rectification on the secondary(MOSFETs used in our design for SR), while L_o and C_o form the output filter. Shim inductor L_R provides assistance to the transformer leakage inductance for resonance operation with MOSFET capacitance and facilitates Zero Voltage Switching (ZVS). The switching waveforms for the system are shown in Figure 4-3.

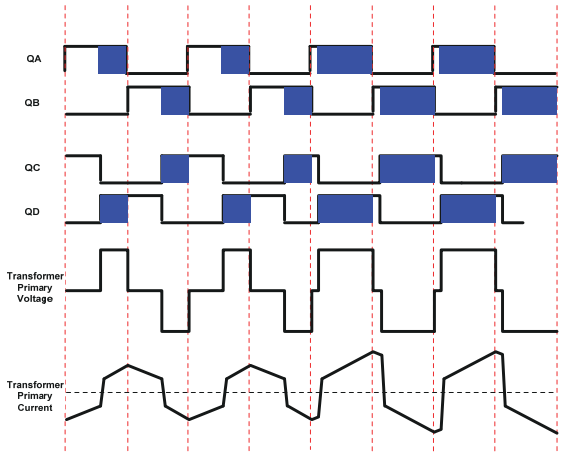


Figure 4-3. PSFB PWM Waveforms

4.3 System Design Theory

4.3.1 Peak Current Mode Control (PCMC) Implementation

Implementing PCMC for a PSFB system requires complex PWM waveform generation with precise timing control. The Piccolo family of devices from Texas Instruments feature advanced on-chip control peripherals that make this implementation possible without any external support circuitry for this purpose. These peripherals include on-chip analog comparators, digital-to-analog converters (DAC), advanced PWM resources and unique programmable on-chip slope compensation hardware. Figure 4-4 shows the PCMC implementation. Transformer primary current is compared with the peak current reference calculated by the voltage loop using the on-chip comparator.

In every half of the switching cycle when the transformer primary current reaches the commanded peak reference value, one of the PWM waveforms driving the switches (Q2/Q3) is *Reset* immediately ending the power transfer phase. The PWM waveform driving the other switch in the same leg is *Set* after a programmable dead-time (dead-band) window. Appropriate slope compensation is also applied that adds a ramp with a programmable negative slope to the peak reference current signal.

The *Resetting* and *Setting* action of the PWMs in one leg results in a phase shift between PWM signals driving the two legs. The amount of this phase shift, and thereby the overlap between diagonal switches, is dependent on the amount of peak reference current. Higher the peak reference current, longer the overlap between diagonal switches, and thereby, more the energy transferred to the secondary. The controller regulates the output by controlling this energy transfer by way of controlling the peak reference current value. Thus this peak reference current is the controlled parameter.

An important feature of this implementation is that the same peak reference current command is used for both halves of the switching cycle under all operating conditions. This provides optimal flux balance for the transformer primary reducing any chances of saturation.

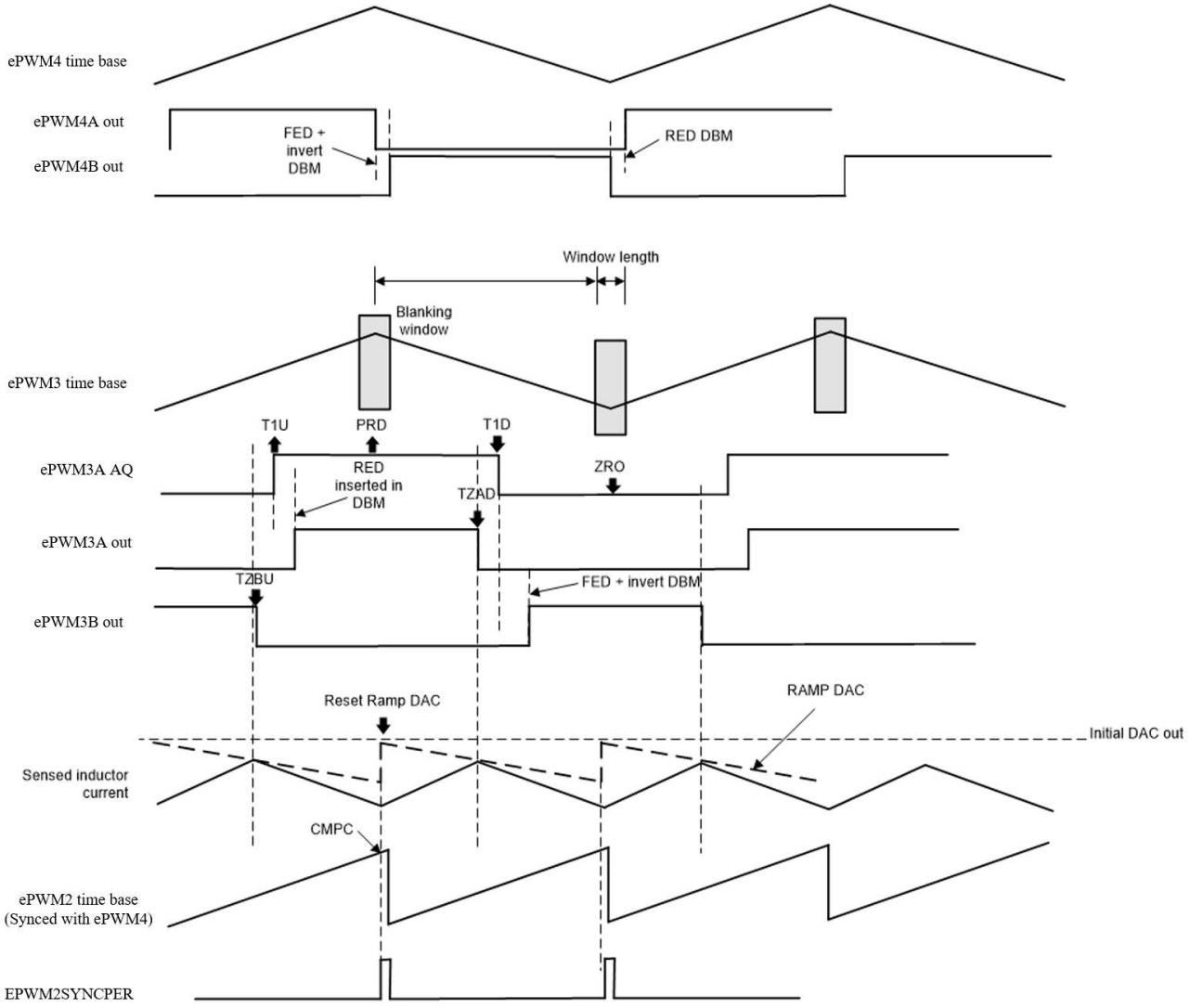


Figure 4-4. PCMC PSFB Implementation

The EPWM4A and EPWM4B outputs are complimentary and 50% duty cycle are maintained based on the action qualifier and deadband settings. The EPWM3A/B_AQ (action qualifier output) is set to 1 when T1U(T1 with counter counting up) event occurs and when counter reaches PRD. The EPWM3A_AQ is set to 0 when T1D (T1 with counter counting down) event occurs and when counter reaches ZERO. The T1D and T1U events are triggered by DCAEVT2.

With type-5 PWM, the deadband can be directly added to T1U and T1D events without adding extra code configurations. The rising edge delay is inserted for EPWM3A. And falling edge delay is inserted for EPWM3B together with inversion accordingly.

To achieve the quickest response, the trip zone modules are utilized to setup the falling edge for EPWM3A (with TZAD event) and EPWM3B (with TZBU event) output. The cycle by cycle trip is implemented and is required to be cleared on ZERO and PRD event.

4.3.2 Zero Voltage Switching (ZVS) or Low Voltage Switching (LVS)

PSFB DC-DC converters make use of parasitic elements in the circuit to provide zero voltage across the MOSFET switches before turning them ON, providing soft switching. This considerably reduces the amount of switching losses associated with hard switching.

For the system discussed here, switching transitions for switches in the Q2- Q3 leg end the power transfer interval. Therefore this leg is called the *Active to Passive* leg. When transitions occur for switches in this leg, current in the primary winding is close to the maximum magnitude for that half PWM switching cycle. The reflected load current aids the circulating energy in the primary circuit during this time, which makes it possible for voltage across switches in this leg to approach zero volts. It is possible to achieve ZVS for switches in this Q2-Q3 leg across the complete load range. Note that as the load decreases the amount of dead-time needs to be increased to achieve and approach ZVS.

Switching transitions for switches in the Q1- Q4 leg start the power transfer interval. Therefore this leg is called the *Passive to Active* leg. During these switching transitions, primary current decreases. It crosses zero current value and changes direction. This results in lower available energy for ZVS. In fact for operations under low load conditions, voltage across these switches may not go to zero before turning them on. Switching losses can be kept to a minimum by turning these switches ON at a time when the voltage across them is at a minimum. This is called Low Voltage Switching or low voltage switching (LVS). As the load changes the time at which the switch should be turned on to achieve LVS changes, requiring dead-time adjustment similar to the Q2-Q3 leg switches.

4.3.3 Synchronous Rectification

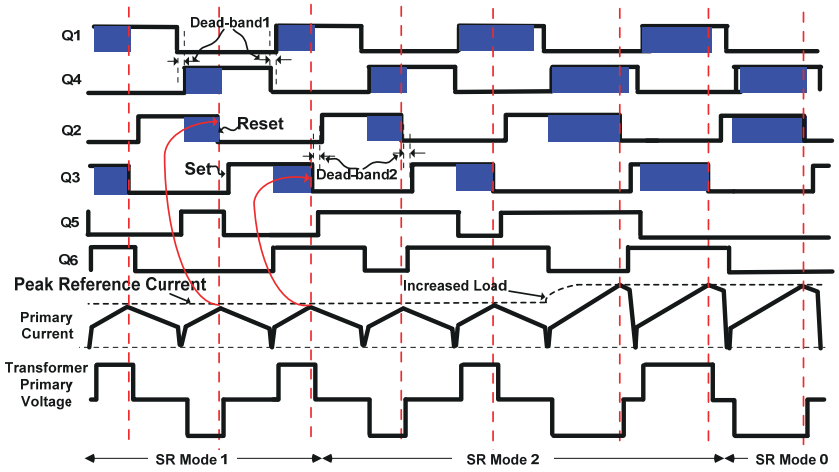


Figure 4-5. Synchronous Rectification Mode

Synchronous rectifiers can work in one of the following three modes at any given time:

- **Mode 0:** This is the classical diode current doubler mode achieved by keeping synchronous rectifiers turned OFF. It is useful for very low load operations where synchronous rectifier switching losses are greater than the power savings obtained by synchronous rectification.
- **Mode 1:** In this mode the synchronous rectifier switches behave like ideal diodes. This mode is useful when operating at very low to low loads, typically when burst mode is being used. In this mode, synchronous rectifier MOSFETs are ON only when the corresponding diagonal bridge drive signals overlap.
- **Mode 2:** Useful for all other load conditions. In this mode, synchronous rectifier MOSFETs are OFF only when the corresponding opposite diagonal bridge drive signals overlap.

Figure 4-5 depict waveforms generated for driving the synchronous rectifier switches in these modes. It is important to implement mode transitions seamlessly without any glitches or anomalies on the PWM outputs even during large load transients or sudden phase shift change commands to provide safe operation of the system.

4.3.4 Slope Compensation

The peak current reference command is written to the RAMPMAXREFS which is the starting value of the RAMP used for slope compensation. The slope is decided by RAMPDACVALS. The ramp generator produces a falling-ramp input for the high reference 12-bit DAC when selected. Note that in this mode, the reference 12-bit DAC uses the most significant 12 bits of the RAMPSTS countdown register as the input. The low 4 bits of the RAMPSTS countdown register effectively act as a prescale for the falling-ramp rate configurable with RAMPDECVALA.

In this reference design, the slope is chosen based on two factors: stability and transient response. The empirical value used in the design should make the system stable and also does not sacrifice the transient response. If the slope is too large, it limits the duty cycle available and decreases the transient response. If the slope is too small, it triggers the sub-harmonics oscillation which creates unstable conditions.

5 Hardware

5.1 Hardware Overview

TIDM-02009 PSFB Converter Hardware shows key hardware components. This hardware uses the kit of Bidirectional 400V/12V DC/DC converter reference design as the base board and with some modification for peak current mode control.

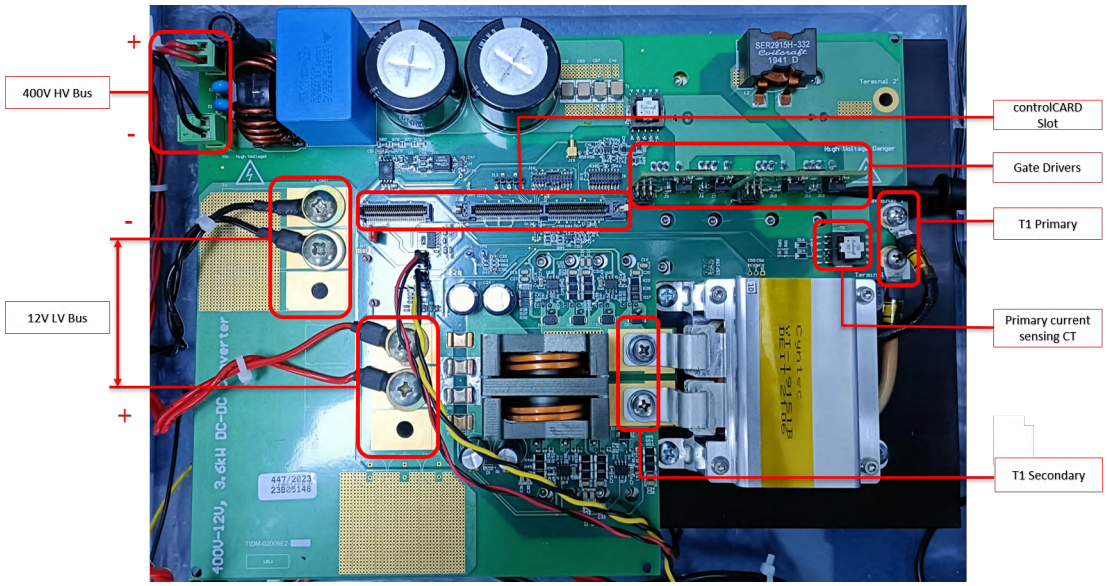


Figure 5-1. TIDM-02009 PSFB Converter Hardware

The key signal connections between the controlCARD and the base board are listed in the Key Signal Connections. The HSEC pin number is based on the E2 version of the AM263x controlCARD (TMDSCNCD263).

Table 5-1. Key Signal Connections

Signal Name	Description	Connection to ControlCARD	HSEC Pin Number
ePWM-4A	PWM drive for full-bridge switch Q1	EPWM4_A	57
ePWM-4B	PWM drive for full-bridge switch Q4	EPWM4_B	59
ePWM-3A	PWM drive for full-bridge switch Q2	EPWM3_A	54
ePWM-3B	PWM drive for full-bridge switch Q3	EPWM3_B	56
ePWM-5A	PWM drive for sync rectifier/push-pull switch Q5	EPWM5_A	61
ePWM-5B	PWM drive for sync-rectifier/ push-pull switch Q6	EPWM5_B	63
VLV-FB	Low voltage bus – voltage feedback	ADC3_AIN2	34
ILV-FILT	Heavily filtered low voltage current feedback	ADC1_AIN2	18
VHV-FB	High voltage bus – voltage feedback	ADC3_AIN1	30
IHV-FILT	Heavily filtered transformer high voltage winding current	ADC4_AIN0 /ADC_CAL0	25

While running the hardware, pin #1 and #2 of J19 in the base board must be shorted with a jumper. 12V power supply must be applied in between pin #9 and#7 of J17 connector only if gating pulses for the Q5, Q6 are to be enabled.

5.2 Hardware and Test Instruments Required

The reference design is set up with the following elements:

- One TIDM-02009 DCDC converter power board

The following list shows the key resources used for controlling the power stage on the MCU:

- Two TIDM-02009 gate driver card
- One TMDSCNCD263 controlCARD
- One 5V, 3A Raspberry Pi adapter
- Mini USB cable
- Computer for loading software to the AM263x MCU

The following list of test equipment is required to power and evaluate the design:

- 15V, 4A bench-style supply for primary board power

For DCDC converter Mode:

- 250W DC electronics load
- 1000V, 5A, 5kW power supply for HVDC link input
- Oscilloscope with current and high-voltage probes

5.3 TMDSCNCD263 controlCARD™ Changes

This change is required only if *LD16* of AM263x controlCARD (TMDSCNCD263) keep on blinking due to insufficient current input. In such case, R340 resistor of the Power MUX circuit of TMDSCNCD263 as shown in ControlCARD Fix for power conflict has to be removed and controlCARD must be powered up by a 5V, 3A power adapter.

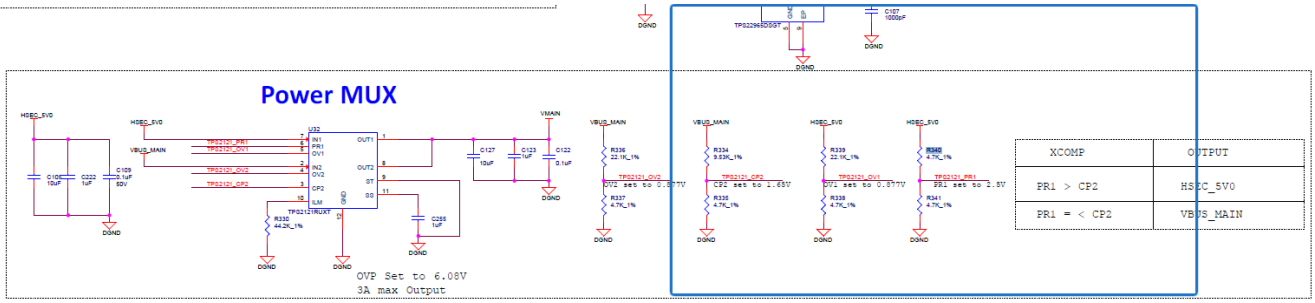


Figure 5-2. ControlCARD Fix for Power Conflict

6 Software

6.1 Getting Started With Firmware

6.1.1 Opening the Code Composer Studio™ Project

The software of this design is available inside the Digital_Power_SDK_AM263x_tirex. Open the project using the following steps:

1. If necessary, install Code Composer Studio™ (CCS) version 12.5 or above by following the instructions on the AM263x MCU+ SDK Getting Started Guide.
2. The minimum supported SysConfig version is 3.2.0 or higher.
3. The minimum supported DIGITAL_POWER_SDK version for this demonstration is digital_power_sdk_am263x_09_01_00_01.
   - Report in the E2E™ forums if the latest DIGITAL_POWER_SDK is not supported for the demonstration.
4. If necessary, the installation steps for each package is provided as part of the SDK and Tools Setup.
5. Download the latest digital power SDK from DIGITAL_POWER_SDK_AM263x download page. Install the SDK to the local PC.
6. Open CCS, and create a new workspace.
7. Inside CCS, go to *Project → Import CCS Project*. Navigate to the location of the firmware, and import the *.\digital_power_sdk_am263x_09_01_00_01\examples\tidm-02009-dcdc\am263x-cc\r5fss0-0_nortos\ti-arm-clang\example.projectspec* .
8. To launch the project using CCS, set the EVM in DevBOOT mode. Follow the steps mentioned in the EVM Setup section of the Getting Started Guide to set up the EVM.

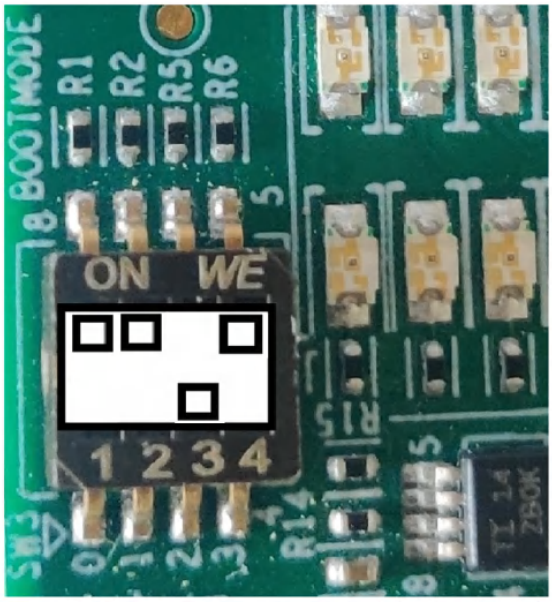


Figure 6-1. DevBoot Mode

9. To load and run the code into the EVM, follow the steps mentioned in the *Load and Run Example* section of the *Getting Started Guide*.

6.1.2 Software Architecture

The project flow contains initialization of all the peripheral clocks and submodules of the system on a chip (SOC) using the AM263x SysConfig Tool and MCU_PLUS_SDK_AM263x. Initializing the Peripheral IPs used in this design like EPWM, ADC, CMPSS and so forth, are discussed in Section 6.2.

The Software contains one ISRs and one background task. The code flow is described in Figure 6-2. The ISR is scheduled to run every 10µs. It contains the main control loop explained in Section 4.3.1.

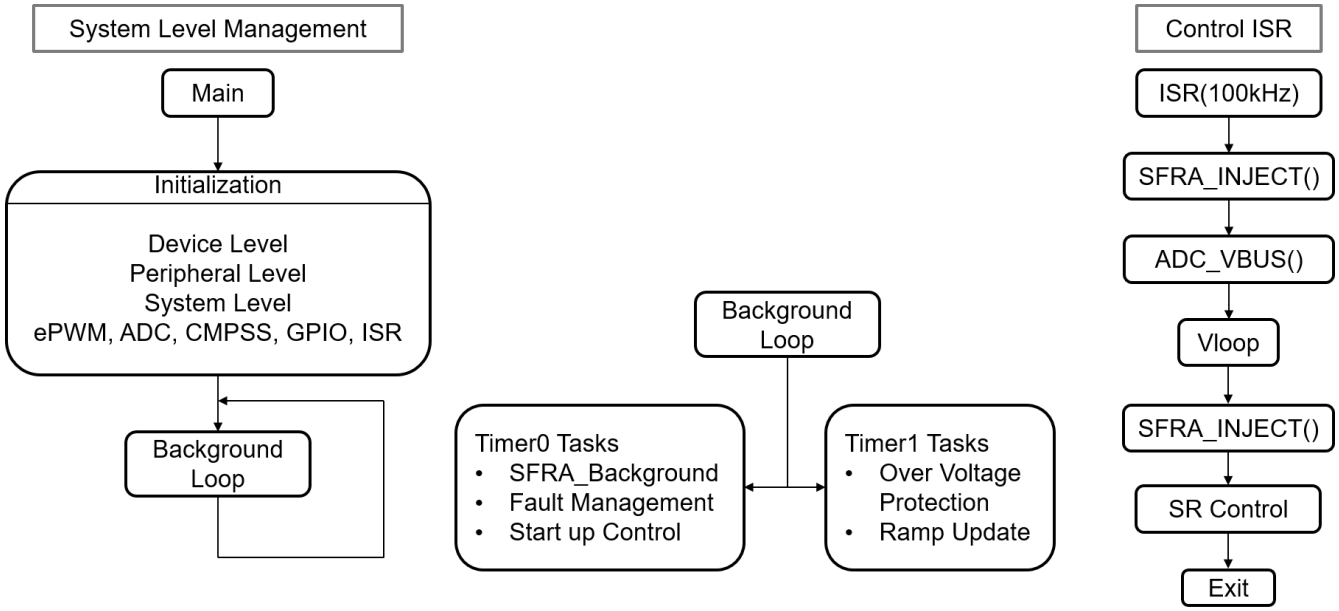


Figure 6-2. Software Block Diagram

6.1.3 Project Folder Structure

Project Structure shows the general structure of the project. Once the project is imported, the *Project Explorer* appears inside CCS.

Design-specific and device-independent files that consist of the core algorithmic code are in the `Source` folder. Library and Control Algorithms related to Digital Power Applications are present in the `libraries` folder. Board specific and Driver specific changes are configured in SysConfig and this code is automatically generated by SysConfig in the `Release/SysCfg` folder. The `main.c` file consists of the main framework of the project. This file consists of calls to the board and file that helps create the system framework, along with the interrupt service routines (ISRs) and slow background tasks. Figure 6-3 explains a detailed folder structure.

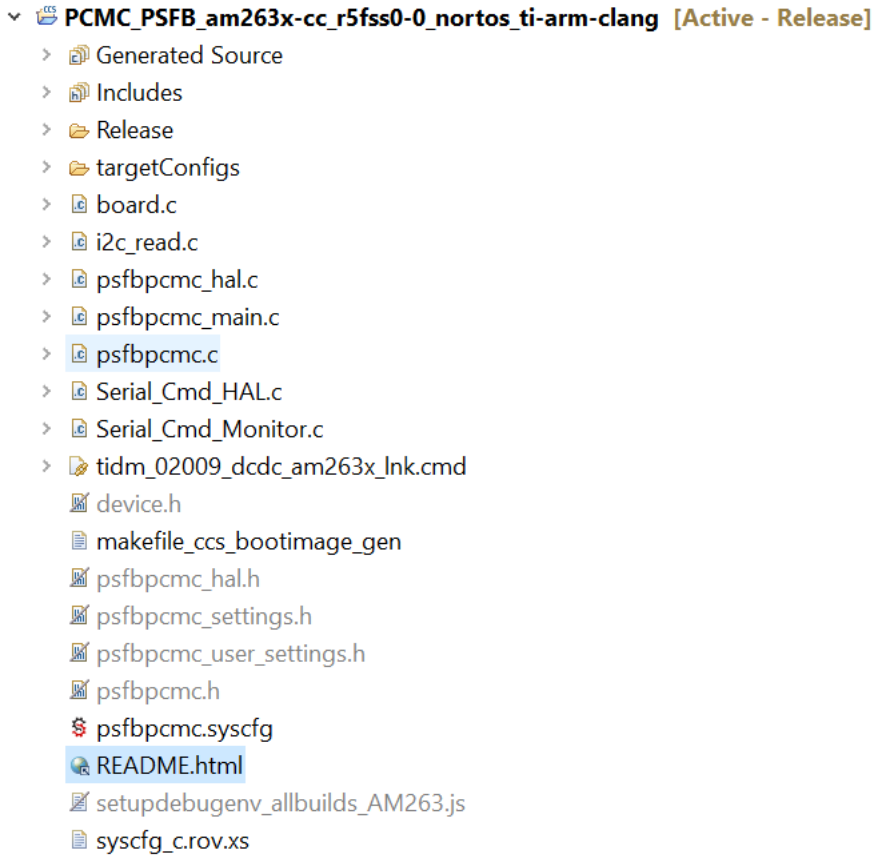


Figure 6-3. Project Structure

All variables and function calls are prepended by the *PSFB* name (for example, *PSFB*_vLVBus_sensed_pu). This naming convention lets the user combine different designs while avoiding naming conflicts.

6.2 SysConfig Setup

The AM263x controlCARD has two released revisions-E1 and E2. The revisions differ in ADC-CMPSS Mappings. Update the .syscfg file according to the board revision in use. This project is developed according to E2 version of the controlCARD.

The psfbpcmc.syscfg is used to configure all the modules present in the PSFB converter control. To run the control loop, the modules needed are the ADC, CMPSS and EPWM modules. Each of these modules are independently explained in the next section.

6.2.1 EPWM Configuration

PWMs are one of the core parts of the software configuration for the inverter. The PWM module of AM263x inherited features from TI classic C28 controllers. A total of six PWM channels are needed to control the inverter output.

Two EPWM instances are created for controlling the two legs of the HV side: *PSFB_FB_PWM1*, *PSFB_FB_PWM2*. *PSFB_SR_PWM* is created to control the SR switches of the LV side. *PSFB_ADC_OVERSAMPLING_PWM* is created to generate EPWMSYNCPER signal form CMPSS module. The EPWM peripheral clock is running at 200MHz. *PSFB_FB_PWM1*, *PSFB_FB_PWM2* and *PSFB_SR_PWM* are configured for 100kHz frequency at Up-Down Count mode. *PSFB_ADC_OVERSAMPLING_PWM* is configured for 200kHz frequency at Up Count mode.

$$\text{Time Base Period for Up} - \text{Down Count Mode} = \frac{\left(\frac{\text{EPWMCLK}}{\text{HSPCLK} \times \text{CLKDIV}}\right)}{2 \times \text{ReqFreq}} = \frac{50\text{MHz}}{100\text{kHz}} = 500 \quad (1)$$

$$\text{Time Base Period for Up Count Mode} = \frac{\left(\frac{\text{EPWMCLK}}{\text{HSPCLK} \times \text{CLKDIV}}\right)}{\text{ReqFreq}} = \frac{100\text{MHz}}{200\text{kHz}} = 500 \quad (2)$$

From Equation 1, Time base Period of *PSFB_FB_PWM1*, *PSFB_FB_PWM2*, *PSFB_SR_PWM* is 500, when the High-Speed Clock divider is 1 and Time Base Clock divider. From Equation 2, the Time base Period of *PSFB_ADC_OVERSAMPLING_PWM* is also 500, when the High-Speed Clock divider is 1 and Time Base Clock divider. Only PWM instance assignment was done using *SysConfig*. *PSFB_FB_PWM1*, *PSFB_FB_PWM2*. *PSFB_SR_PWM* and *PSFB_ADC_OVERSAMPLING_PWM* have been assigned to EPWM4, EPWM3, EPWM5 and EPWM2, respectively. All the other PWM related configurations as discussed in Section 4.3.1 and Section 4.3.3 are done in *PSFB_HAL_setupPwms()* inside *main()*.

6.2.2 ADC Configuration

Four ADC instances have been used in this demonstration to sense all voltages, currents on the HV and LV dc bus side. *PSFB_HVBUS_ADC_MODULE, PSFB_LVBUS_ADC_MODULE, PSFB_IHV_FILT_ADC_MODULE, PSFB_ILV_ADC_MODULE* are assigned to the signals mentioned in Key Signal Connections. Rest of the required ADC related configurations are done in *PSFB_HAL_setupADC()* and *PSFB_HAL_setupADCSoC()* inside *main()*.

The ADC HSEC Board Pin out is different for E1 and E2 versions of the AM263x controlCARD, which is mentioned in Table 6-1.

Table 6-1. ADC Mapping for E1 and E2 controlCARD With HSEC Board

HSEC Board	E1	E2
12	ADC0_AIN0	ADC1_AIN0
14	ADC0_AIN1	ADC1_AIN1
15	ADC0_AIN2	ADC0_AIN2
18	ADC1_AIN0	ADC1_AIN2
20	ADC1_AIN1	ADC1_AIN3
21	ADC1_AIN2	ADC0_AIN4
23	ADC1_AIN3	ADC0_AIN5
28	ADC2_AIN2	ADC3_AIN0
30	ADC2_AIN3	ADC3_AIN1
31	ADC3_AIN0	ADC2_AIN0
33	ADC3_AIN1	ADC2_AIN1
34	ADC3_AIN2	ADC3_AIN2
37	ADC4_AIN0	ADC2_AIN2
39	ADC4_AIN3	ADC2_AIN3

Set the ADC DAC reference voltage switches according to application requirements by checking in the respective board schematics document in the section - ADC and DAC Interfaces.

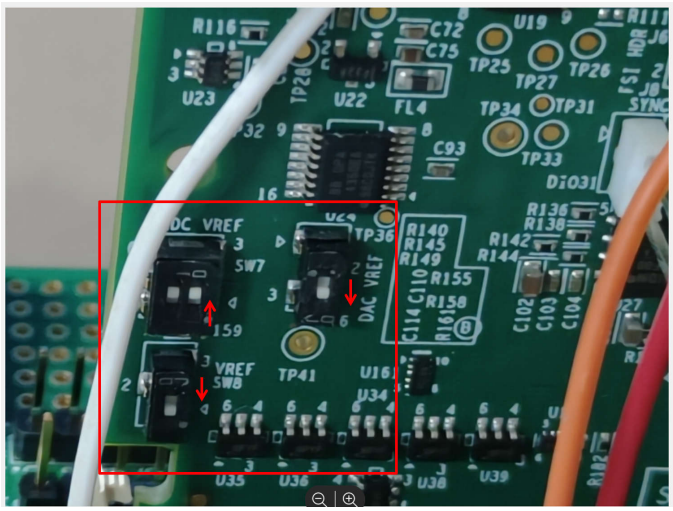


Figure 6-4. ADC and DAC Reference Switches in the AM263x controlCARD

6.2.3 CMPSS Configuration

Required CMPSS configuration as mentioned in Section 4.3.4 has been done using *SysConfig*. *PSFB_PCMC_OCP_CMPSS* is assigned to *CMPSSA8* and the required configurations are shown in Figure 6-5.

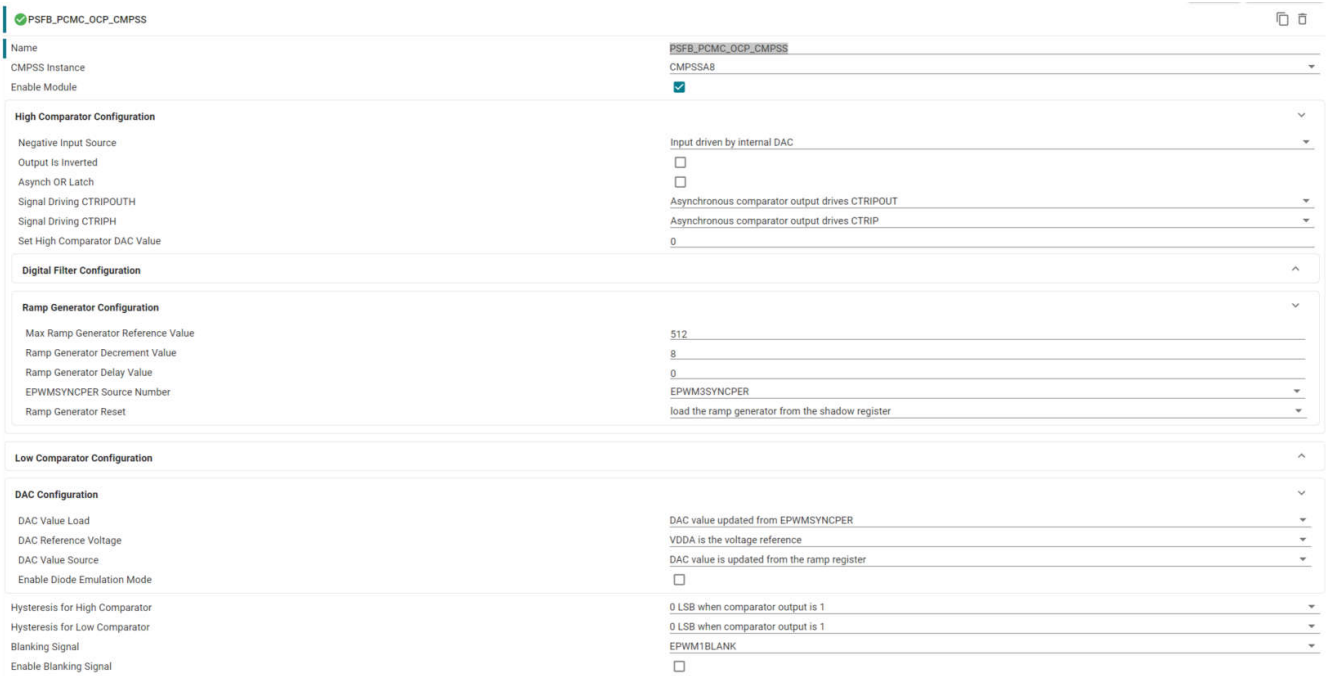


Figure 6-5. CMPSS Configuration

6.3 Incremental Builds

Testing high-voltage reference designs is usually performed in different stages beginning with open loop tests in low voltage where board protection is disabled. In this stage the basic functionality – for example, interrupt and task switching, ADC sensing, PWM outputs, and so on – are tested without going deep into the control loop functions. Once the basic functionality is established, board protection logics are introduced to validate the high-voltage functionality. As the voltage increases, the noise in the system increases, hence the software needs to retains the functionality even in high-noise conditions. Simultaneously the software needs to trip the PWMs in case of any undesired overvoltage or overcurrents in the system that degrade or destroy the design. The next stage involves closing the loop to test the full functionality of the design.

The software is designed to switch between these stages for ease of testing and validating the firmware. The following sections describe what are the objectives of these stages and how to switch between the stages.

6.3.1 Procedure for Running the Incremental Builds - PCMC

6.3.1.1 Lab 1: Phase Overlapping Check With Open Current and Voltage Loop

6.3.1.1.1 Objective of Lab 1

The objective of this build is to control the power flow by manually adjusting the phase shift between two legs, verify the DAC and ADC driver modules, verify the MOSFET driver circuit and sensing circuit on the board and become familiar with the operation of Code Composer Studio (CCS). Since this system is running open-loop, the ADC measured values are only used for instrumentation purposes in this build. Steps required to build and run a project are explored.

6.3.1.1.2 Overview of Lab 1

The software in Build 1 is configured so that the user can quickly evaluate the DAC driver module by viewing various waveforms on an oscilloscope and observe the effect of change in the phase shift value reference command on the output voltage by interactively adjusting this command from CCS. Additionally, the user can evaluate the ADC driver module by viewing the ADC sampled data in the watch view. The ePWM4 module acts as the master time-base for the system. It operates in up-down count mode. ePWM4A and ePWM4B drive Q1 and Q4 full-bridge switches, while ePWM3A and ePWM3B drive Q2 and Q3 full-bridge switches. Whenever the phase shift value changes, the time base phase shift register of the ePWM3 module gets updated accordingly and the outputs (both ePWM3A and ePWM3B) start generating pulses with the same amount of phase delay with respect to the ePWM4 module.

6.3.1.1.3 Procedure of Lab 1

6.3.1.1.3.1 Start CCS and Open a Project for Lab 1

To quickly execute this build, follow the following steps:

1. Make sure that all jumpers on the board are correctly installed or removed as listed in the Hardware section.
2. Hardware Set-up: Insert the AM263x controlCARD in the 100-pin DIMM connector. Connect a 5V DC bench power supply between pin number 1 and 3 of J17 connector with appropriate polarity. Short pin number 1 and 2 of J19 connector using a jumper. Connect an isolated 400V programmable DC power source to the 400V input connector and 12V electronic load to the 12V connector (make sure that this load does not exceed the board ratings). Connect a USB cable between the PC and controlCARD. Do not turn ON any of the power supplies at this time.
3. Open Code Composer Studio. Maximize Code Composer Studio to fill your screen. Close the welcome screen if it opens up.
4. Locate the solution package downloaded and import the project in CCS.

6.3.1.1.3.2 Build and Load the Project for Lab 1

1. Click on the expand sign (>) on the left of the project name *PCMC_PSFB_am263x-cc_r5fss0-0_nortos_ti-arm-clang* and open the psfbpcmc_settings.h as shown in Lab 1 setup. The setting is #define INCR_BUILD 1. This enables the build 1 solution.
2. Set #define PSFB_PCMC 0 in the psfbpcmc_settings.h. The default value of PSFB_PCMC is defined as 1 to execute closed current loop (this mode is discussed in detail in upcoming sections).
3. Click on the *View* tab on the top left. Click on *Target Configuration*. The *Target Configuration* window opens. Right click on *user defined* and select *Import Target Configuration*. Browse to the CCS workspace where the project was imported. Open the `targetConfigs` folder and select the `AM263x_Real_Time_debug.ccxml`.
4. Double click on `AM263x_Real_Time_debug.ccxml` and do the *Test Connection* check. If the check fails, change the COM Port number following the steps mentioned in AM263x Real Time Debug setup. This COM Port number varies from one controlCARD to another. Proceed further only when *Test Connection* is successfully passed.
5. Turn ON the 5V DC bench power supply. Connect the USB cable to the Control card and make sure LD7 is glowing green. In the *Target Configuration* window, select AM263x_Real_Time_debug.ccxml. Click on the *Debug* button. The Lab 1 code is built successfully without any error.
6. Notice the *CCS Debug* icon in the upper right corner indicating that the process is now in the *Debug Perspective* view. Right click on the *Cortex_R5_0* in the *Debug* window and select *Connect Target*. Now click on the *Load* icon from the tab on the top side. Load the .out file which was just built successfully. Now .out is successfully loaded to Core_0. The program is stopped at the start of *main()*.

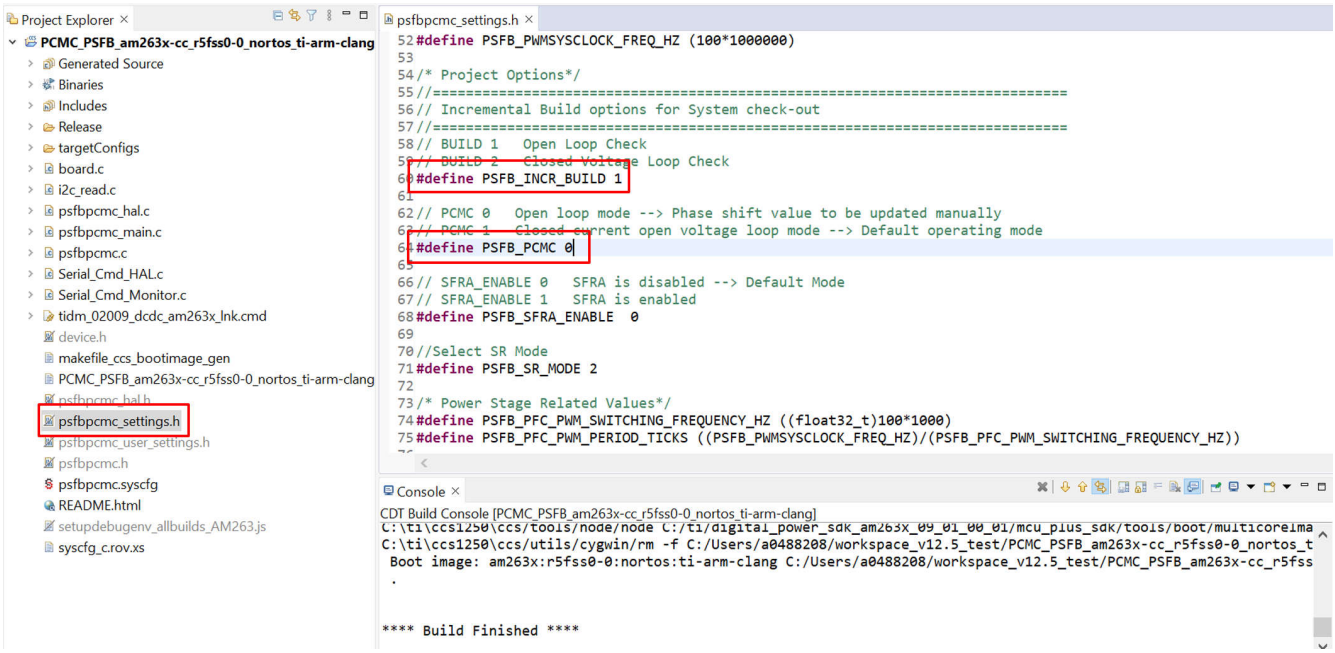


Figure 6-6. Lab 1 Setup

6.3.1.1.3.3 Debug Environment Windows for Lab 1

Populate the expressions window entries by clicking *scripting console* on the menu bar and then opening the *setupdebugenv_allbuilds_AM263.js* file from the project workspace directory (PCMC_PSFB_am263x-cc_r5fss0-0_nortos_ti-arm-clang\setupdebugenv_allbuilds_AM263.js) using the scripting console *Open File* ( ) command. Please note that some of the variables have not been initialized at this point in the main code and may contain some garbage values.

6.3.1.1.3.4 Run the Code for Lab 1

1. Run the code by using the <F8> key, or using the *Run* button on the toolbar.
2. Make sure the *IO Expander found* message is coming in the console as shown in Console message.

3. Follow the *Launch Real Time Debug* of AM263x real time Debug guide to enable real time monitoring of the expression window.
4. Apply an appropriate resistive load to the PSFB system at the DC output. A load that draws around 3A to 6A current at 12V output is a good starting point.
5. Set PSFB_enable to be 1. This enables PWM output. The expression window is similar to Expression Window.
6. Power the input at J1, J2 with 300V DC. Observe the *PSFB_Phase_Shift_Val* parameter in expression window to have a value of 50 at starting (value of 500 indicates 100% phase overlap). Update the value 200. The output is around 6.8V (5A constant current load).
7. Increase the power input from 300V to 400V. Increase the constant current loading to 12A. Slowly increase PSFB_Phase_Shift_Val to be 300 in incremental steps of 25. Carefully observe the output voltage (PSFB_guiLVbus_Volts), the voltage should gradually increase to 12V. This should not be allowed to exceed the capabilities of the board. Different waveforms, like the PWM gate drive signals, input voltage and current and output voltage can 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.
8. Fully stopping the MCU when in real-time mode is a two-step process. With the 400V DC input turned off wait a few seconds. First, stop the processor by using the STOP button on the toolbar. Then turn off the power supply of the MCU.

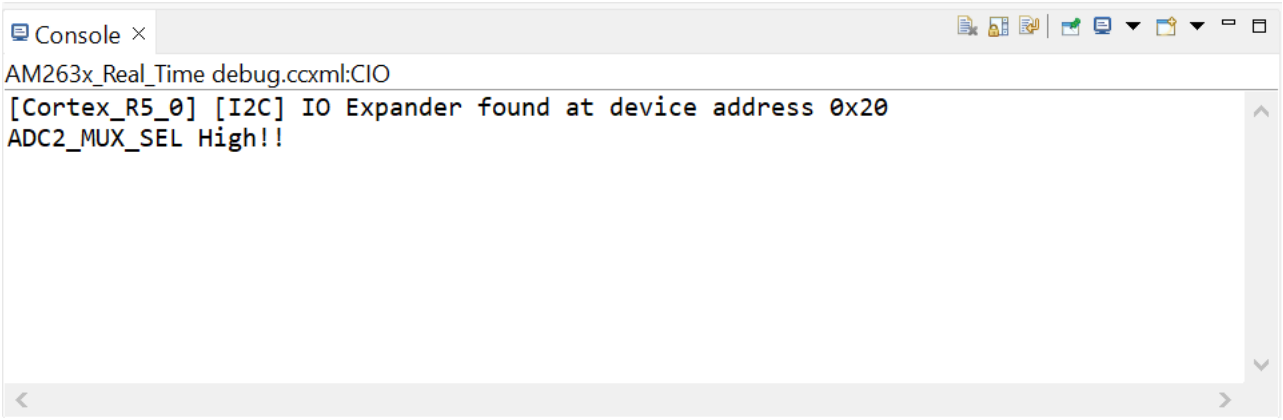


Figure 6-7. Console Message

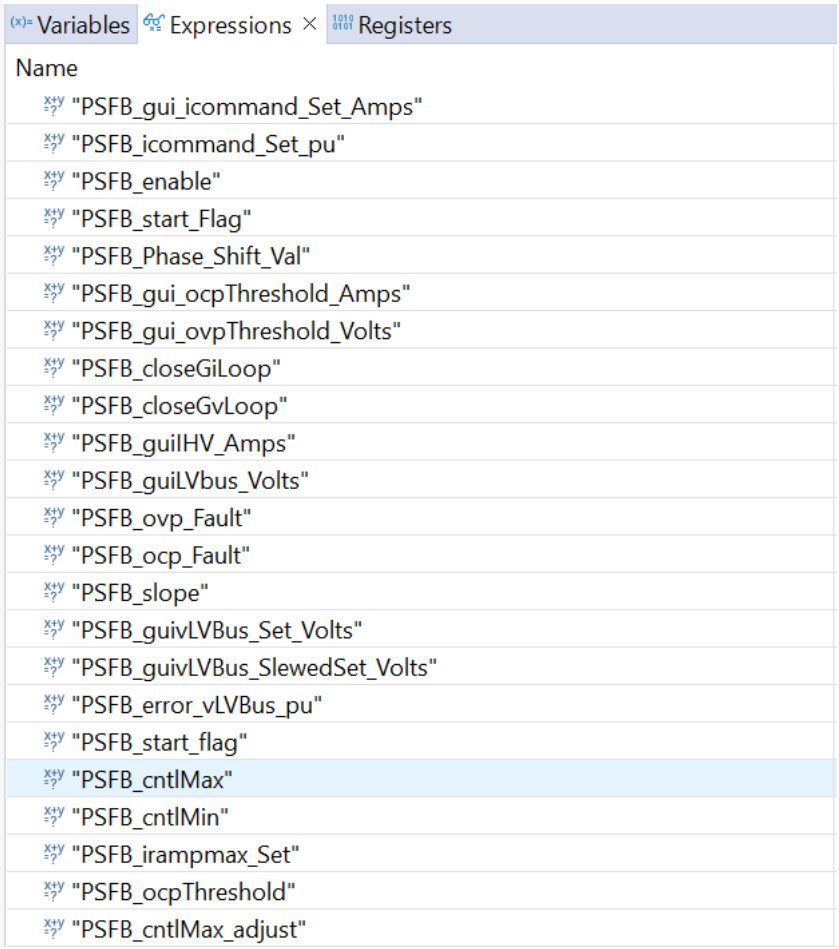


Figure 6-8. Expression Window

6.3.1.2 Lab 2: Closed Current and Open Voltage Loop

6.3.1.2.1 Objective of Lab 2

The objective of this build is to evaluate the peak current mode operation of the system by providing peak current limit as an user input using the CCS environment.

6.3.1.2.2 Overview of Lab 2

The software in lab 2 was configured so that the user can quickly evaluate the DAC driver module by viewing various waveforms on an oscilloscope and observing the effect of change in peak current reference command on the output voltage by interactively adjusting this command from CCS. The on-chip analog comparator compares the transformer primary current with the slope compensated peak current reference. Comparator output is connected to the trip zone logic of the PWM modules. ePWM4 module acts as the master time-base for the system. It operates in up-down count mode. ePWM4A and ePWM4B drive Q1 and Q4 full-bridge switches, while ePWM3A and ePWM3B drive Q2 and Q3 full-bridge switches. ePWM3A and ePWM3B drive Q5 and Q6 synchronous rectifier switches. Although the gating pulses for Q5 and Q6 are active, these switches do not start to operate unless 12V power supply is given to the base board through pin number 9 of J17 port. Whenever the comparator output goes high in a PWM half cycle, the ePWM3 module output (ePWM3A or ePWM3B) which was high at that instant is immediately pulled low while the other PWM3 module output is pulled high after an appropriate dead-band window (dbAtoP). ePWM5A and ePWM5B outputs are driven in a similar way. Note that this slope compensation ramp generation, comparator action and PWM waveform generation are all hardware generated without any software involvement as shown in Figure 6-9. Some register reconfigurations for SR are done inside the ISR.

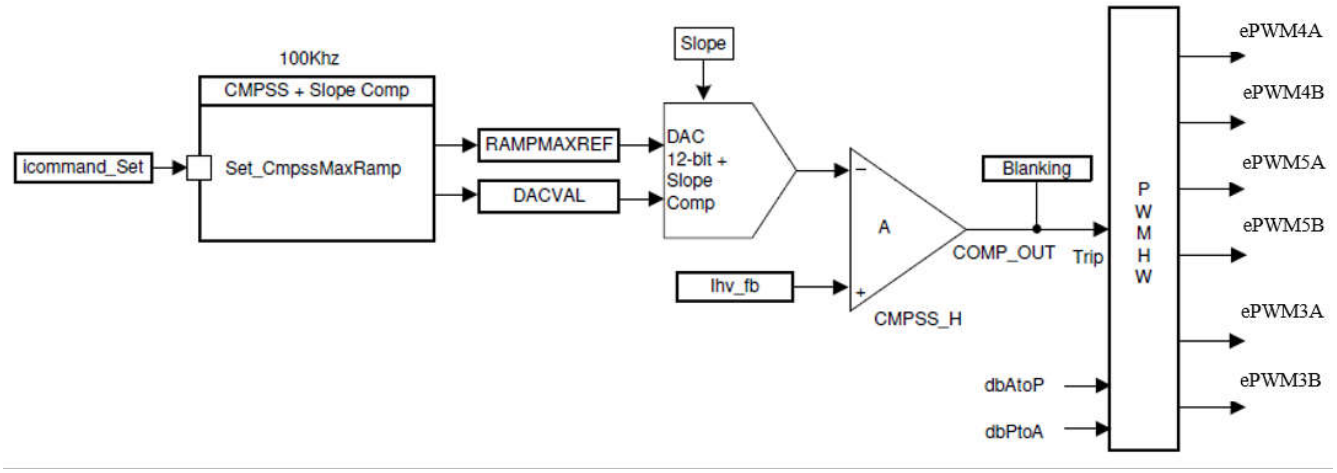


Figure 6-9. Lab 2 Software Blocks

6.3.1.2.3 Procedure of Lab 2

6.3.1.2.3.1 Build and Load Project for Lab 2

To quickly execute this build using the pre-configured work environment, use the following steps:

1. Click on the expand sign (>) on the left of the project name *PCMC_PSFB_am263x-cc_r5fss0-0_nortos_ti-arm-clang* and open the psfbpcmc_settings.h as shown in Lab 2 setup. The setting is #define INCR_BUILD 1. Set #define PSFB_PCMC 1 in the psfbpcmc_settings.h.
2. Turn ON the 12V DC bench power supply. In the *Target Configuration* window, select AM263x_Real_Time_debug.ccxml. Click the *Debug* button. The lab 2 code compiles and loads.
3. Notice the *CCS Debug* icon in the upper right side indicating that the process is now in the *Debug Perspective* view. The program is stopped at the start of *main()*.

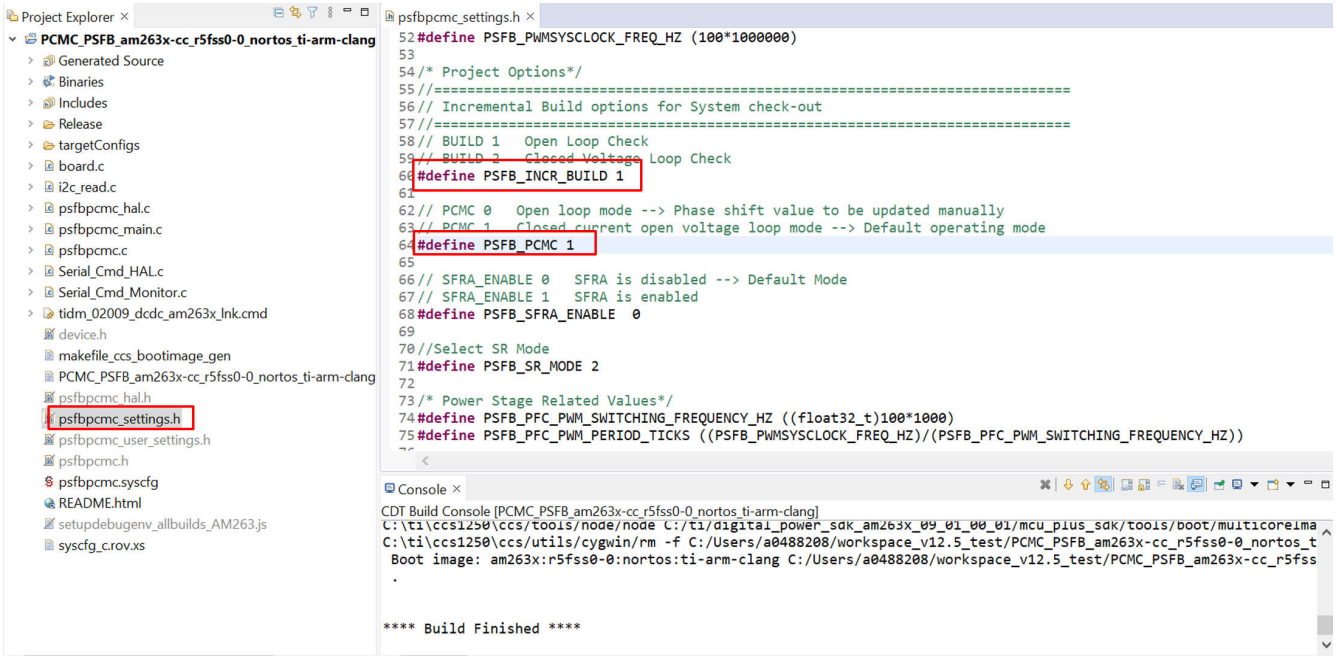


Figure 6-10. Lab 2 Setup

6.3.1.2.3.2 Debug Environment Windows for Lab 2

Populate the expressions window entries by clicking *scripting console* on the menu bar and then opening the *setupdebugenv_allbuilds_AM263.js* file from the project workspace directory (PCMC_PSFB_am263x-

cc_r5fss0-0_nortos_ti-arm-clang\setupdebugenv_allbuilds_AM263.js) using the scripting console *Open File* ( ) command. Please note that some of the variables have not been initialized at this point in the main code and may contain some garbage values.

6.3.1.2.3.3 Run the Code for Lab 2

1. Run the code by using the <F8> key, or using the Run button on the toolbar.
2. Follow the 'Launch Real Time Debug' of AM263x real time Debug guide to enable real time monitoring of the expression window.
3. Apply an appropriate resistive load to the PSFB system at the DC output. A load that draws around 3A to 6A current at 12V output is a good starting point.
4. Set PSFB_gui_icommand_Set_Amps to be 2.3. Set PSFB_enable to be 1.
5. Power the input at J1, J2 with 400V DC. Observe the voltage across the primary side of the transformer using oscilloscope.
6. Slowly increase PSFB_gui_icommand_Set_Amps to be 2.85 in increments of 0.05. Carefully observe the output voltage(PSFB_guiLVbus_Volts), the voltage should gradually increase to 12V. This should not be allowed to exceed the capabilities of the board. 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.
7. The PSFB_slope is set to be 8. User can gradually increase or decrease it for testing.
8. Fully stopping the MCU when in real-time mode is a two-step process. With the 400V DC input turned off wait a few seconds. First, stop the processor by using the STOP button on the toolbar. Then turn off the power supply of the MCU.

6.3.1.3 Lab 3: Closed Current and Closed Voltage Loop

6.3.1.3.1 Objective of Lab 3

The objective of this build is to verify the operation of the complete PCMC based HVPSFB project from the CCS environment.

6.3.1.3.2 Overview of Lab 3

Lab 3 Software blocks shows the software blocks used in this build. A two pole two zero controller is used for the voltage loop. As shown in Lab 3 Software blocks, the voltage loop block is executed at 100kHz. The DCL_runClamp_C1 function is used to avoid controller wind-up condition. The DCL_runDF22_C2 computes the immediate part of the precomputed DF22 controller. If this value is larger than the max or less than min threshold, a flag is set with DCL_runClamp_C1 function. This stops the further controller saturation. Once the flag is off, DCL_runDF22_C3 computes the partial result of the pre-computed DF22 controller. More information on how the DCL function works is found inside the DCL user's guide in C2000ware.

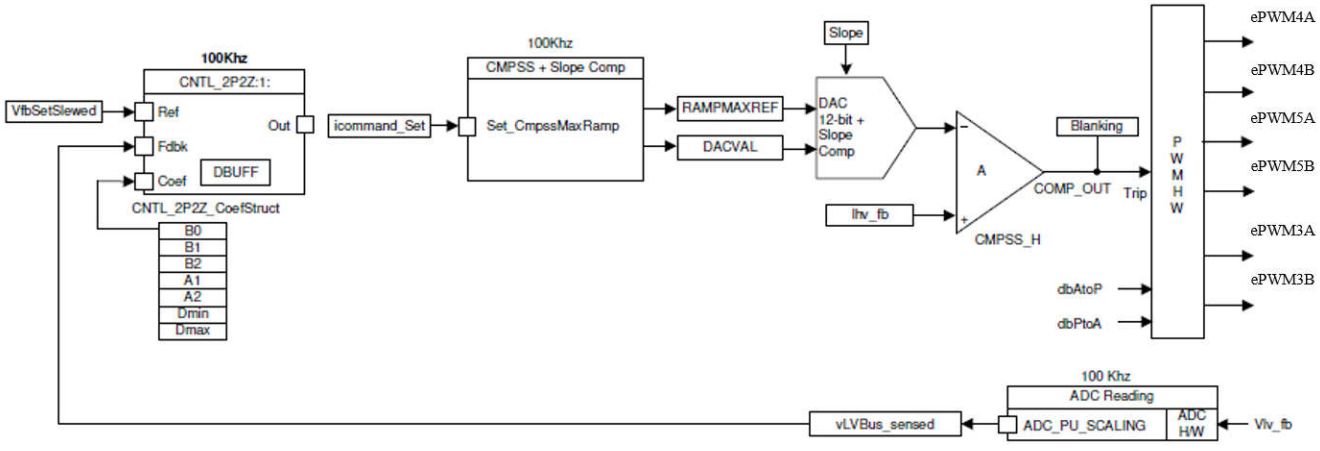


Figure 6-11. Lab 3 Software Blocks

6.3.1.3.3 Procedure of Lab 3

6.3.1.3.3.1 Build and Load Project for Lab 3

1. Click on the expand sign (>) on the left of the project name *PCMC_PSFB_am263x-cc_r5fss0-0_nortos_ti-arm-clang* and open the psfbpcmc_settings.h as shown in Figure 6-12. Set #define INCR_BUILD 2 in the psfbpcmc_settings.h.
2. Turn ON the 12V DC bench power supply. In the *Target Configuration* window, select `AM263x_Real_Time_debug.ccxml`. Click the *Debug* button. Lab 3code compiles and loads.
3. Notice the *CCS Debug* icon in the upper right side indicating that the process is now in the *Debug Perspective* view. The program is stopped at the start of *main()*.

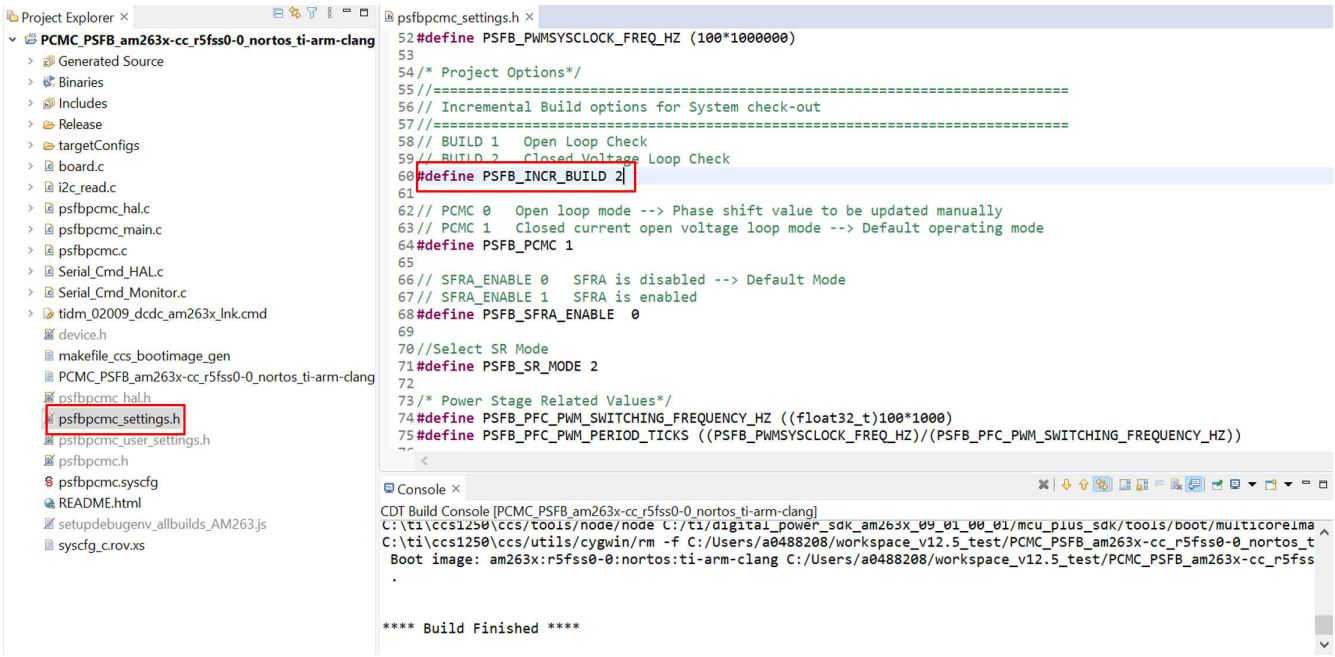


Figure 6-12. Lab 3 Setup

6.3.1.3.3.2 Debug Environment Windows for Lab 3

Populate the expressions window entries by clicking *scripting console* on the menu bar and then opening the *setupdebugenv_allbuilds_AM263.js* file from the project workspace directory (PCMC_PSFB_am263x-cc_r5fss0-0_nortos_ti-arm-clang\setupdebugenv_allbuilds_AM263.js) using the scripting console *Open File* ( ) command. Please note that some of the variables have not been initialized at this point in the main code and can contain some garbage values.

6.3.1.3.3.3 Run the Code for Lab 3

1. Follow Steps 1 to 4 of the lab 2 procedure to enable real-time mode and continuous refresh for the watch views and also for changing the continuous refresh interval for the watch view if needed.
2. Apply an appropriate load to the PSFB system at the DC output. A load that draws around 3A–6A current at 12V output is a good starting point.
3. Check PSFB_guivLVBus_Set_Volts. PSFB_guivLVBus_Set_Volts is targeted to be 2 at the beginning.
4. Power the input at J1, J2 with 400V DC (200V and 300V are optional). The output voltage needs to rise up to 2V.
5. Set PSFB_start_flag to 1. The output voltage now starts ramping up to 12V. Here is the watch view that corresponds to the operation of the system with 12V at the output with an input voltage of around 400V and a load of around approximately 5A (constant current load).
6. Gradually increase the load (constant current load) to 20A. The output power is around 250W.
7. By default, the synchronous rectifiers are operated in mode 2. You can change the mode of operation by changing PSFB_SR_mode variable to 0, 1, or 2 from the watch view. Observe the change in the amount of input current being drawn and changed in output voltage with different SR modes. Probe the PWM

waveforms driving the synchronous rectifier switches. Do not change between different SR modes when operating at very low loads or when the output voltage is very low (less than 6V). In these cases use the default SR mode 2. To enable gating pulses for Q5 and Q6, provide 12V, 0.75A supply between pin #10 and #8 of the J17 connector.

7 Testing and Results

7.1 Lab 0: Basic PWM Check

Lab 0 can be run on the AM263x controlCARD and docking station. To execute Lab 0, build the code by following the steps mentioned in Section 6.3.1.1.3.2.

Connect the ADC4_AIN0, ADC1_AIN1, ADC3_AIN1, ADC3_AIN2 pins of the controlCARD docking station to ground point. Connect the EPWM3, EPWM4, EPWM5 pins of the controlCARD docking station to a logic analyzer to view the PWM waveforms.

- ADC4_AIN0 – Pins 25
- ADC1_AIN1 – Pins 18
- ADC3_AIN1 – Pins 30
- ADC3_AIN2 – Pins 32
- EPWM3 – Pins 54, 56
- EPWM4 – Pins 57, 59
- EPWM5 – Pins 61, 63

Figure 7-1 shows the expected PWM waveforms.

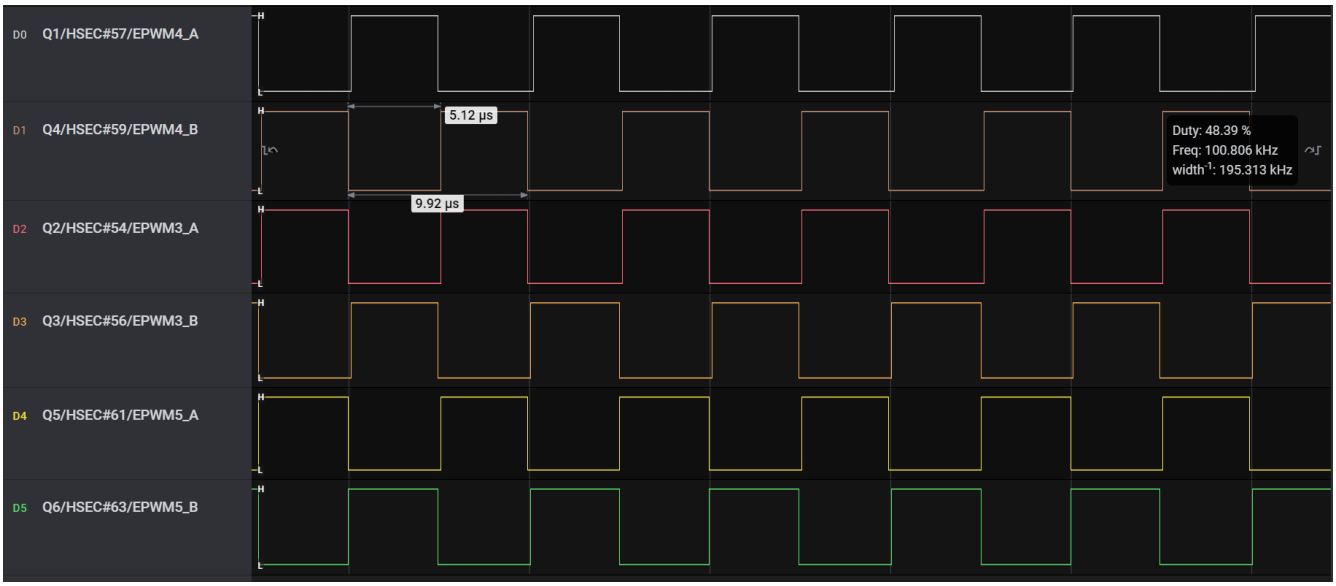
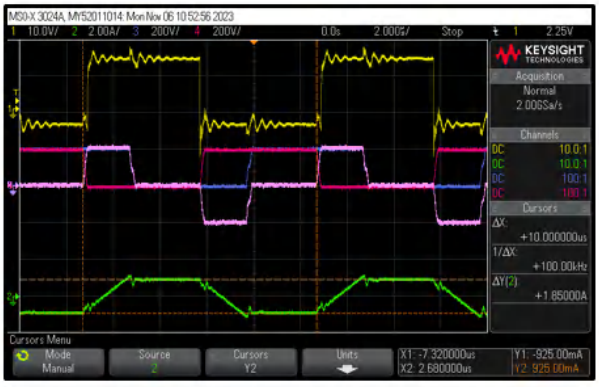


Figure 7-1. PWM Waveforms

7.2 Lab 1: Phase Overlapping Check With Open Current and Voltage Loop

Proceed to execute Lab 1 only after making sure all the waveforms are coming as per the results of Lab 0 shown in PWM waveforms.

Execute Lab 1 by placing the controlCARD in the base board of the power converter. Execute this lab exercise following the steps mentioned in Section 6.3.1.1.3. Two sets of waveforms are presented in Set 1 and Set 2. Waveforms in yellow, red, blue, pink, green show voltage across gate to source of Q4, voltage across drain to source of Q3, voltage across drain to source of Q4, voltage across primary side of the transformer and transformer primary current, respectively.



Gating pulse across Q4, Transformer primary Voltage and Current at 200V, 13W

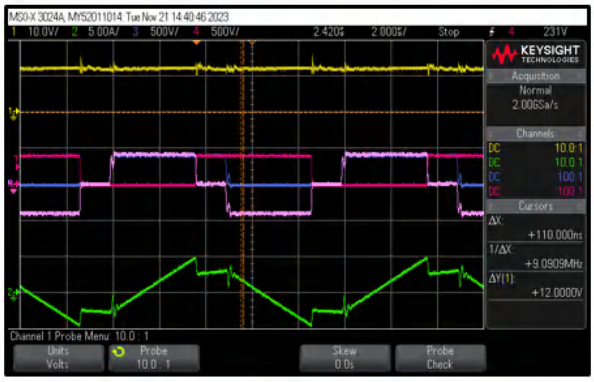
Figure 7-2. Testing at 200V, 13W

Gating pulse across Q4, Transformer primary Voltage and Current at 400V, 120W

Figure 7-3. Testing at 400V, 120W

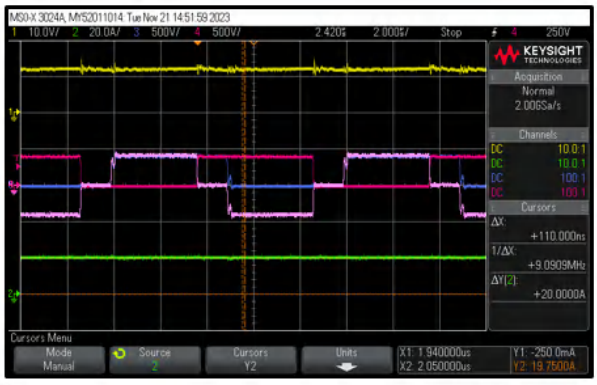
7.3 Lab 2: Closed Current and Open Voltage Loop

Execute the Lab 2 exercise following the steps mentioned in Section 6.3.1.2.3. Two sets of waveforms are presented in Set 1 and Set 2. Waveforms of Set 1 in yellow, red, blue, pink, green show voltage across the load, voltage across drain to source of Q3, voltage across drain to source of Q4, voltage across primary side of the transformer and transformer primary current, respectively. Waveforms of Set 2 in yellow, red, blue, pink, green show voltage across the load, voltage across drain to source of Q3, voltage across drain to source of Q4, voltage across primary side of the transformer and load current, respectively.



Output Voltage, Transformer primary Voltage and Current at 400V, 246W

Figure 7-4. Set 1

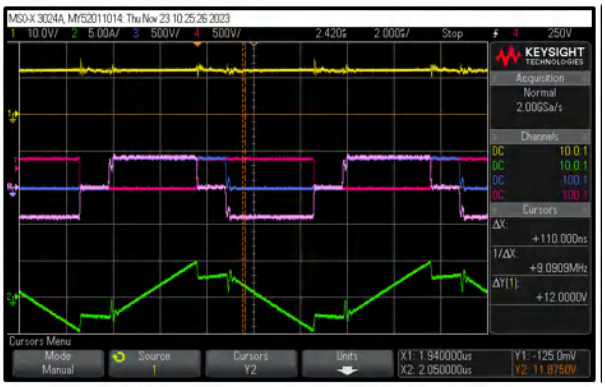


Output Voltage, Transformer primary Voltage and Output Current at 400V, 246W

Figure 7-5. Set 2

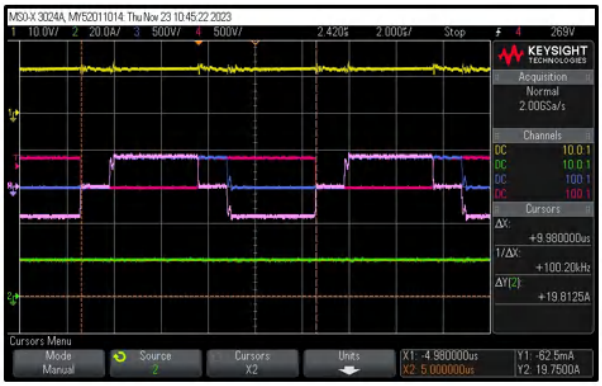
7.4 Lab 3: Closed Current and Closed Voltage Loop

Execute the Lab 3 exercise following the steps mentioned in Section 6.3.1.3.3. Two sets of waveforms are presented in Set 1 and Set 2. Waveforms of Set 1 in yellow, red, blue, pink, green show voltage across the load, voltage across drain to source of Q3, voltage across drain to source of Q4, voltage across primary side of the transformer and transformer primary current, respectively. Waveforms of Set 2 in yellow, red, blue, pink, green show voltage across the load, voltage across drain to source of Q3, voltage across drain to source of Q4, voltage across primary side of the transformer and load current, respectively.



Output Voltage, Transformer primary Voltage and Current at 400V, 248W

Figure 7-6. Set 1



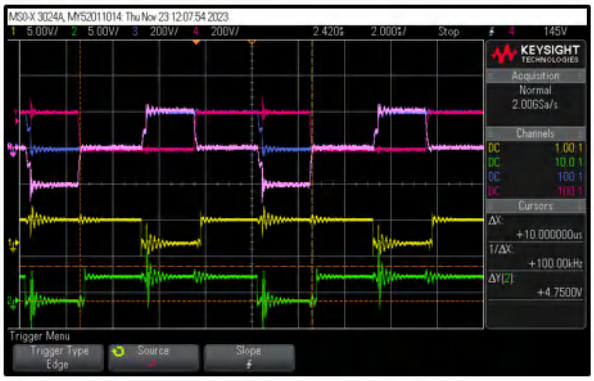
Output Voltage, Transformer primary Voltage and Output Current at 400V, 248W

Figure 7-7. Set 2

After enabling the gating pulses of Q5 and Q6, waveforms shown in SR mode enabled at 120W and SR mode enabled at 240W can be obtained. In both of them, waveforms of in red, blue, pink, yellow, green show voltage across drain to source of Q3, voltage across drain to source of Q4, voltage across primary side of the transformer, voltage across drain to source of Q5 and voltage across drain to source of Q6, respectively.

Transformer primary Voltage, Voltage across Q5 and Q6 at 120W

Figure 7-8. SR Mode Enabled at 120W



Transformer primary Voltage, Voltage across Q5 and Q6 at 240W

Figure 7-9. SR Mode Enabled at 240W

8 References

- Texas Instruments, *TIDM-02000 Peak current-mode controlled phase-shifted full-bridge reference design using C2000™ real-time MCU*
- Texas Instruments, *TIDM-02009 ASIL D safety concept-assessed high-speed traction, bi-directional DC/DC conversion reference design*
- Texas Instruments, *DIGITAL POWER SDK-AM263x Digital Power SDK for AM263X*

9 Revision History

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

Changes from Revision * (January 2024) to Revision A (April 2024)	Page
• Updated software download paths	11
• Updated Figure 6-3	12
• Updated Figure 6-6	17
• Updated Figure 6-10	20
• Updated Figure 6-12	22

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