AN_2003_PL88_2006_221026

The XDPP1100 digital power supply controller

XDPP1100 application note

About this document

Scope and purpose

This document introduces the XDPP1100 digital controller for isolated power supplies. The major features and benefits of telemetry sensing, loop control and protections are described. Special features such as feed-forward (FF), current sharing, flux balancing, non-linear fast transient response (FTR) and burst mode operation are also introduced and demonstrated with experiment results.

Intended audience

Power supply design engineers, isolated digital brick module designers, telecom and server power system designers.

Table of contents

1 General introduction

The XDPP1100 is a digital power supply controller based on a 32-bit, 100 MHz ARM® Cortex™-M0 RISC microprocessor with analog/mixed-signal capabilities, on-chip memory and communication peripherals. The device is specifically optimized to enhance the performance of isolated DC-DC applications and reduce the solution component count in the IT and network infrastructure space.

The XDPP1100 is available in VQFN-40 and VQFN-24 packages; both packages have 0.5 mm pitch for ease of manufacturing. **Figure 1** shows the XDPP1100 product offering.

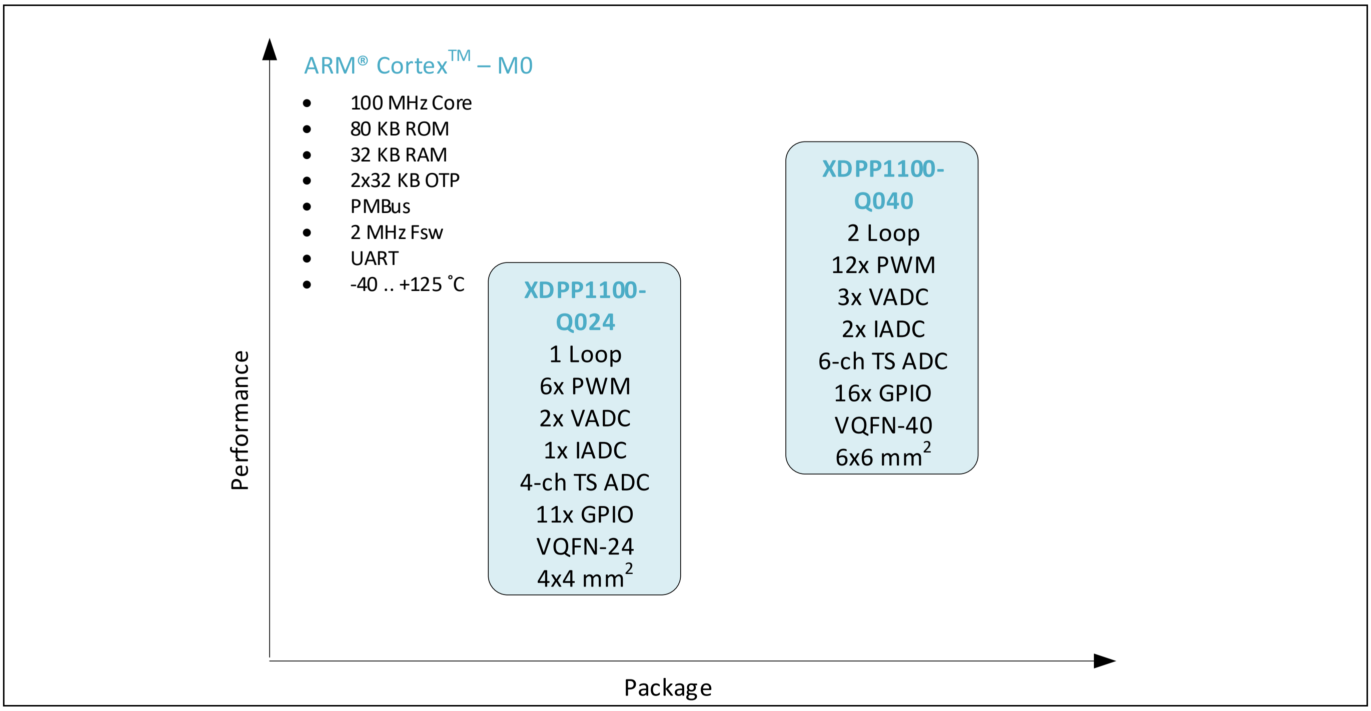
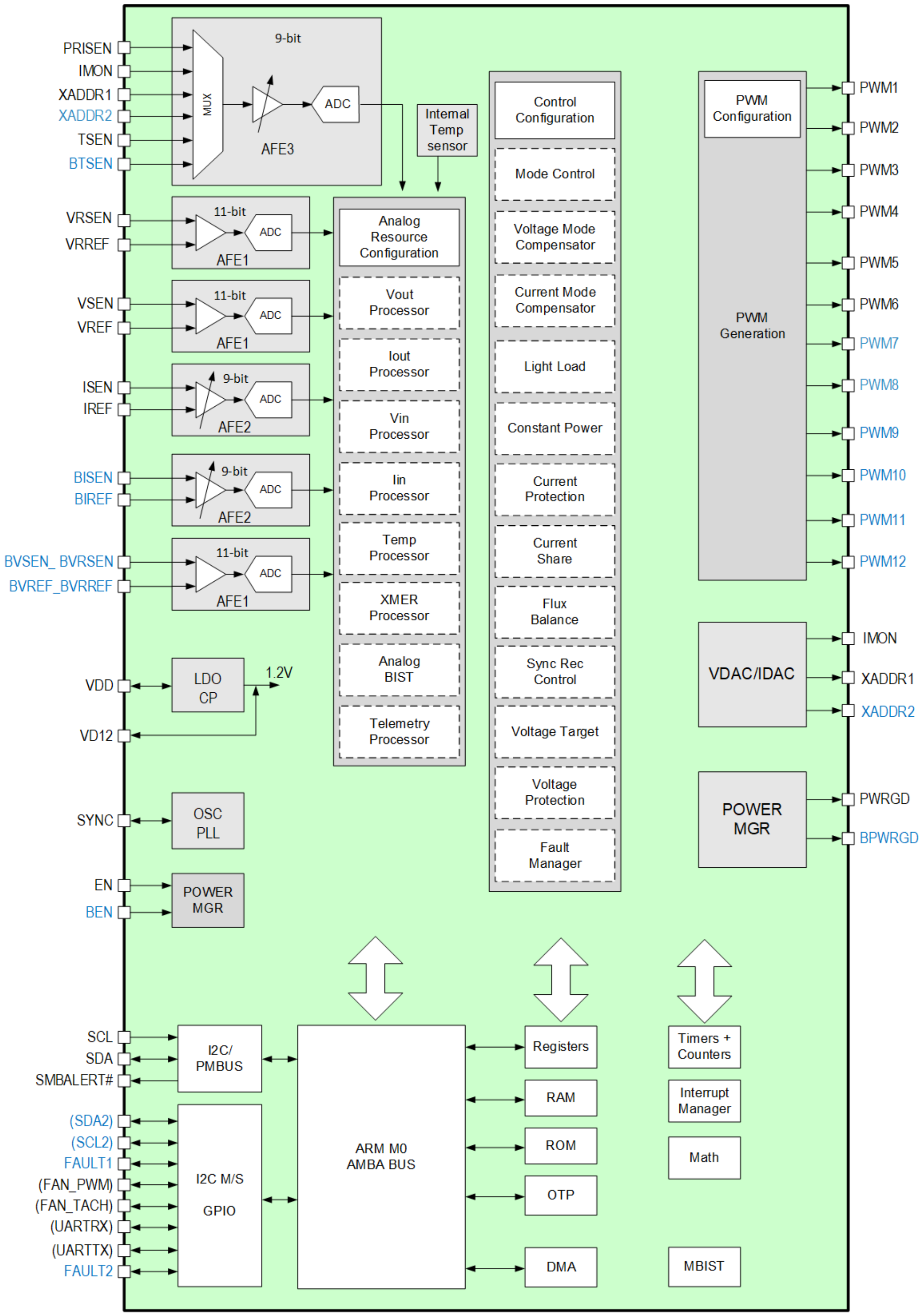


Figure 1 XDPP1100 product offerings

The XDPP1100 supports up to two high-speed digital control loops. Each loop consists of a dedicated voltage analog-to-digital converter (VADC), high-resolution current ADC (IADC), a proportional, integral, derivative (PID)-based digital compensator, and DPWM outputs with 78.125 ps pulse width (PW) resolution. The device also offers six channels of 9-bit, 1 Msps general-purpose ADCs (TS ADCs), timers, interrupt control, PMBus and two I²C communication ports.

The block diagram of the XDPP1100 is shown in **Figure 2**. It contains three high-speed high-resolution VADCs (AFE1), two current sense (CS) ADCs (AFE2) with current ripple estimator, one general-purpose ADC (AFE3) with six input channels to digitize voltage, resistance and temperature, followed by digital voltage processors and current processors. The signals are processed and sent to the digital core. The digital control loop is state machine-based and an ARM® CORTEX M0 core is integrated into the XDPP1100. This architecture makes it fully programmable with high-bandwidth (BW) performance. The pulse width modulation (PWM) block has two ramp generators and 12 pulse generators for up to 12 PWM outputs.

The communication is achieved by I²C and PMBus protocol. The GPIO pins can be configured to enable, power good, sync, fault monitor, fan control, or UART serial port.



XXX: pins that are available in XDPP1100-Q040K64

Figure 2 Block diagram

1.1 Hardware-enabled features

The XDPP1100 has the following hardware (HW) enabled features:

- PID compensation filter
- Current sharing using IMON pin
- Current balancing of interleaved phases
- Feed-forward
- Flux balancing
- Fast load-transient response
- Multi-segment droop
- Burst operation
- Diode emulation (DE) mode
- Adaptive SR dead-time
- Sync-in/sync-out
- Fault protections
- Common fault report and protection
  - Flux balance fault
  - Voltage sense open-loop fault
  - Cycle-by-cycle peak current limit (PCL)
  - Short-circuit protection (SCP)

The details of each feature are described in individual application notes.

1.2 Firmware-enabled features

The firmware (FW) build in the ROM of the IC is pre-programmed to support topology-dependent features, start-up and shutdown sequence and ramp control, telemetry, fault management and reporting. The FW is identical in both 24-pin and 40-pin packages. The XDPP1100 has the following FW-enabled features:

- Start-up and shutdown timing control
- DE mode control
- Telemetry gain and offset adjustment
- Telemetry resolution configuration
- Fault configuration and fault mask
- Temperature compensation for copper trace current shunt
- GPIO configuration
- Allows user patch for extended control and functions. For example:
  - User-defined temperature lookup table
  - Dead-time adjustment per lookup table
  - Fan speed control
  - Frequency dithering
  - Post-soldering trim
  - Snapshot
  - UART configuration
  - Frequency sweep during start-up ramp

- Added droop during V_{OUT} ramp
- Accurate input current and input power telemetry based on efficiency matrix

1.3 VADC

The XDPP1100 has up to three high-speed VADCs for input and output voltage sensing: VSEN/VREF, VRSEN/VRREF and BVSEN_BVRSEN/BVREF_BVRREF. The pin name with "REF" indicates this is the reference input of the differential pair.

Key features of the VADC:

- 100 MHz 11-bit ADC with 3-bit modulation (1.25 mV/least significant bit (LSB), 156 µV resolution)
- 0 to 2.1 V differential voltage range
- Differential sensing for each channel
- Set-point accuracy three-sigma limit is +/-1 percent overtemperature -40 ~ +125°C
- 200 MHz edge detection comparator
- Adaptive step size
- Low-latency protection comparators for overvoltage protection (OVP) and undervoltage protection (UVP) (50 ns)

Key benefits of the VADC:

- Accurate output voltage set-point
  - 156 µV reference resolution
  - 16-bit VOUT_SCALE_LOOP resolution
- Accurate output voltage over process, stress and temperature range
- High-speed sensing over full-scale voltage range
  - Senses switching waveform from transformer secondary side for primary voltage sensing, eliminates isolation device
  - Excellent feed-forward performance
  - Enables volt-second balancing for full-bridge (FB) converter
  - Enables dynamic SR dead-time optimization
- Eliminates external OVP/UVP comparators

The ADC is clocked at 100 MHz and a voltage sense front-end amplifier processes the voltage at 50 MHz. The VSEN/VREF and VRSEN/VRREF are available in both package options. For a single-loop converter, it is suggested to use VSEN/VREF for output voltage sensing and use VRSEN/VRREF for input voltage sensing. **Figure 3** is an example of an isolated converter using VSEN to sense output voltage, and using VRSEN to sense transformer secondary winding V_{RECT}, thus indirectly sensing input voltage.

For the output voltage sensing, the recommended VADC differential voltage range is 0.8 to 2.1 V. Choose the resistor divider to keep VSEN in this range for the best accuracy.

For a power supply with wide range input voltage, typically input voltage ratio could be 2:1 or even 3:1. For example, input voltage of a telecom brick converter could be 36 to 75 V. Choose the resistor divider for VRSEN carefully to keep the scaled VRSEN within the recommended range over input voltage range. Details of V_{RECT} sensing are described in chapter 2.3.

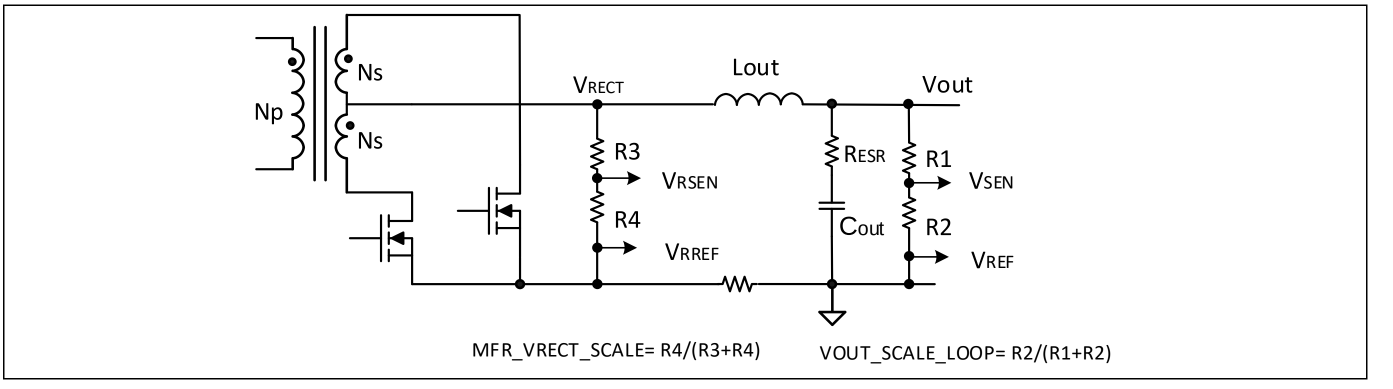


Figure 3 **Input and output voltage sensing of isolated converter**

The third VADC, named BVSEN_BVRSEN/BVREF_BVRREF, is available in the 40-pin package XDPP1100-Q040. It can be used to sense the output voltage of the second loop (BVSEN) in a dual-loop application, or to sense the V_{RECT} of the second phase in an interleaved converter (BVRSEN).

In an interleaved application, sensing the input voltage of the V_{RECT} of the second phase provides more sample data and offers quicker line-transient response. **Figure 4** shows an example of an interleaved half-bridge (HB) converter.

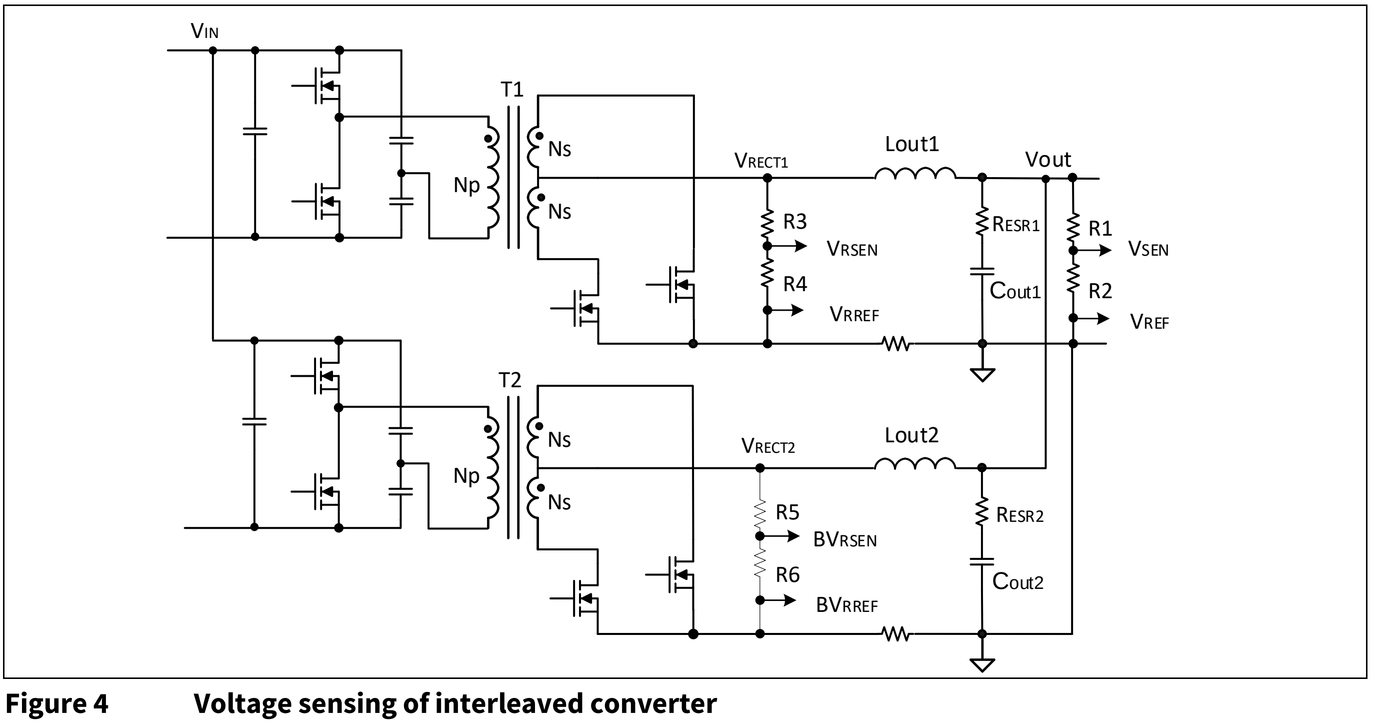


Figure 4 **Voltage sensing of interleaved converter**

In a dual-loop application, the third VADC should be used to sense the output voltage of the second loop. The second loop could be post-buck of an isolated converter or could be an independent converter taking the same input voltage as the first loop. **Figure 5** is an example of post-buck. The second output (V_{out2}) is sensed by BVSEN/BVREF. The input voltage of the post-buck is the output voltage of the first loop (V_{out1}).

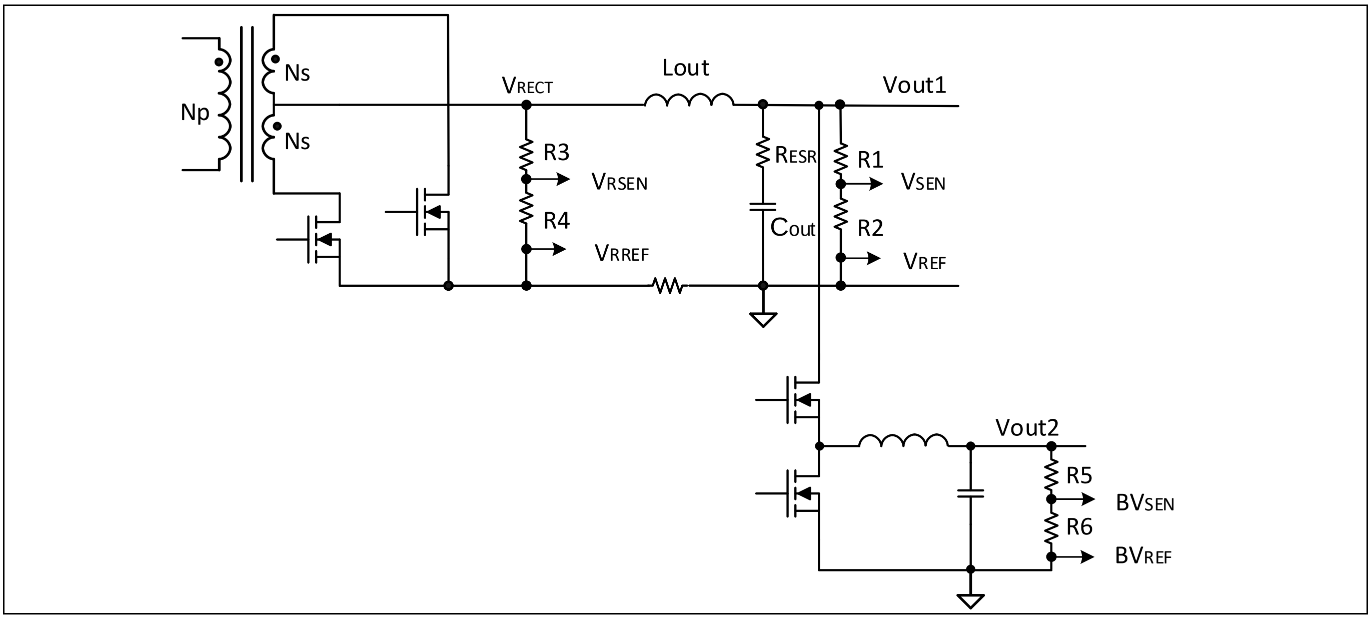


Figure 5 **Dual-loop example: an isolated converter with post-buck**

Figure 6 is an example of dual HB converters. V_{out1} and V_{out2} are independently controlled. V_{out1} and V_{out2} could be the same voltage (i.e. 12 Va, 12 Vb) or different voltages (i.e. 12 V and 5 V). In this example, the second loop obtains the input voltage information from the first loop through the same VRSEN/VRREF ADC. For proper input voltage telemetry, protection and feed-forward compensation, loop 1 must be enabled before loop 2 and must be disabled after loop 2.

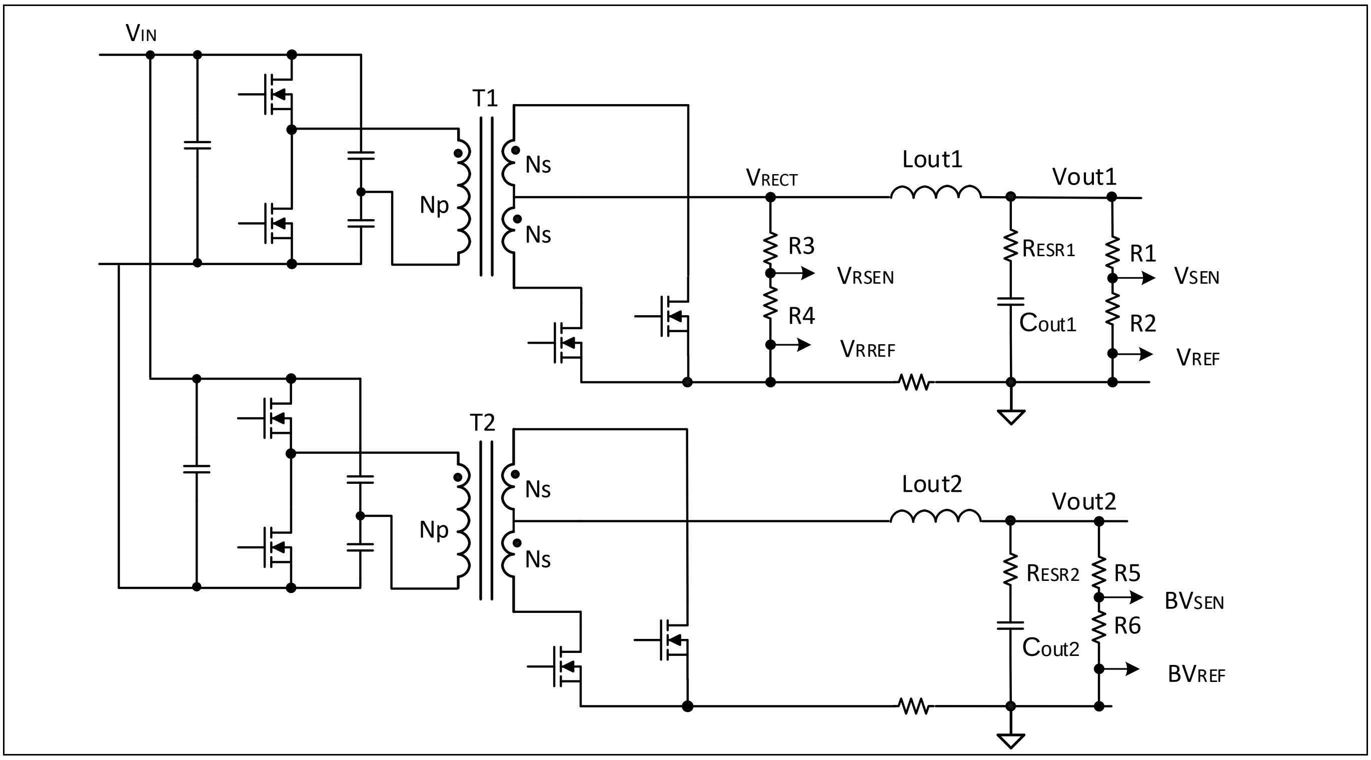


Figure 6 **Dual-loop example: two independent loops sharing the same input**

1.4 IADC

The XDPP1100 has up to two high-speed CS ADCs (IADCs) for current sensing: ISEN/IREF, BISEN/BIREF.

Key features of the IADC:

- 25 MHz 9-bit ADC with selectable gain from 100 µV/LSB, and 1.45 mV/LSB
- Auto-calibrated offset
- Supports CS of integrated power stage
- Low latency protection comparators for over-current protection (OCP) and positive/negative PCL
- Differential sensing for each channel
- Supports temperature compensation

Key benefits:

- Accurate current sense down to 100 µV/LSB
  - Eliminates external high-precision op-amp when using copper trace as current shunt
- Multiple levels of OCP
- Cycle-by-cycle PCL and SCP
  - Eliminates external OCP comparator

The CS ADC (ISEN, BISEN) is 9-bit tracking ADC and is interpolated to 13-bit through current emulation. Detailed information on current sensing is available in chapter 3. The high resolution of IADC enables the XDPP1100 to sense output current through a very small PCB copper shunt, reducing power loss and saving on the cost of the precision sense resistor and op-amp. **Figure 7** is an example of sensing output current by PCB copper trace.

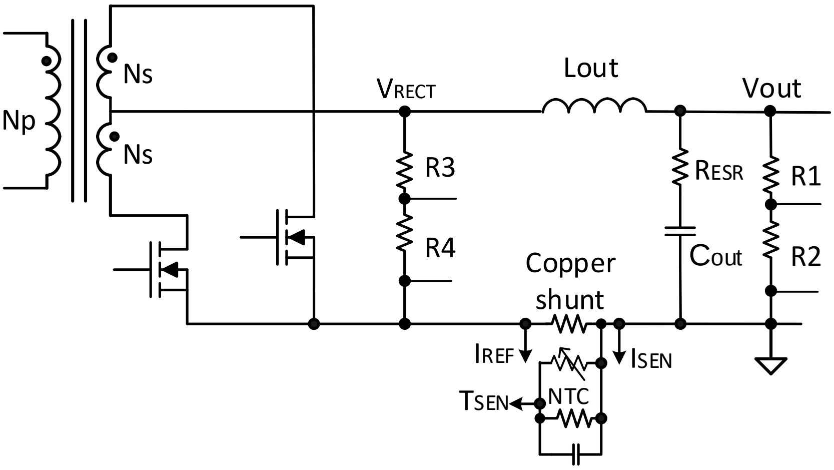


Figure 7 Output current sense with temperature compensation

The 40-pin version XDPP1100-Q040 has two IADCs. This enables dual-loop or dual-phase current sense. It also enables sensing both primary current and secondary current in a single-loop application. Sensing primary current is usually required in primary peak current mode control (PCMC). For isolated converters, this can be done through a CS transformer. **Figure 8** is an example of a FB converter with primary PCMC. It should sense primary current with ISEN ADC and sense the output current with BISEN ADC.

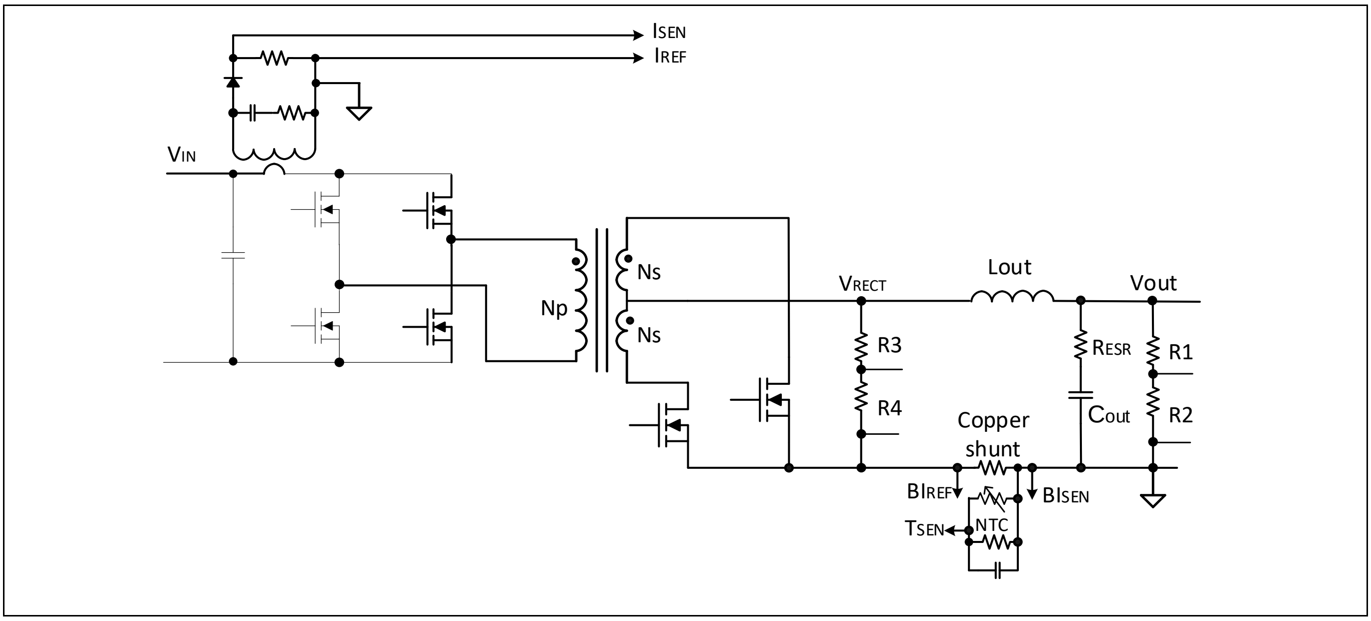


Figure 8 FB converter with primary current sense

In the interleaved topologies, the second IADC (BISEN) is used to sense the load current in the second phase. With both phases current, the converter could calculate total output current and balance the current in each phase. **Figure 9** is an example of interleaved active clamp forward (ACF) topology.

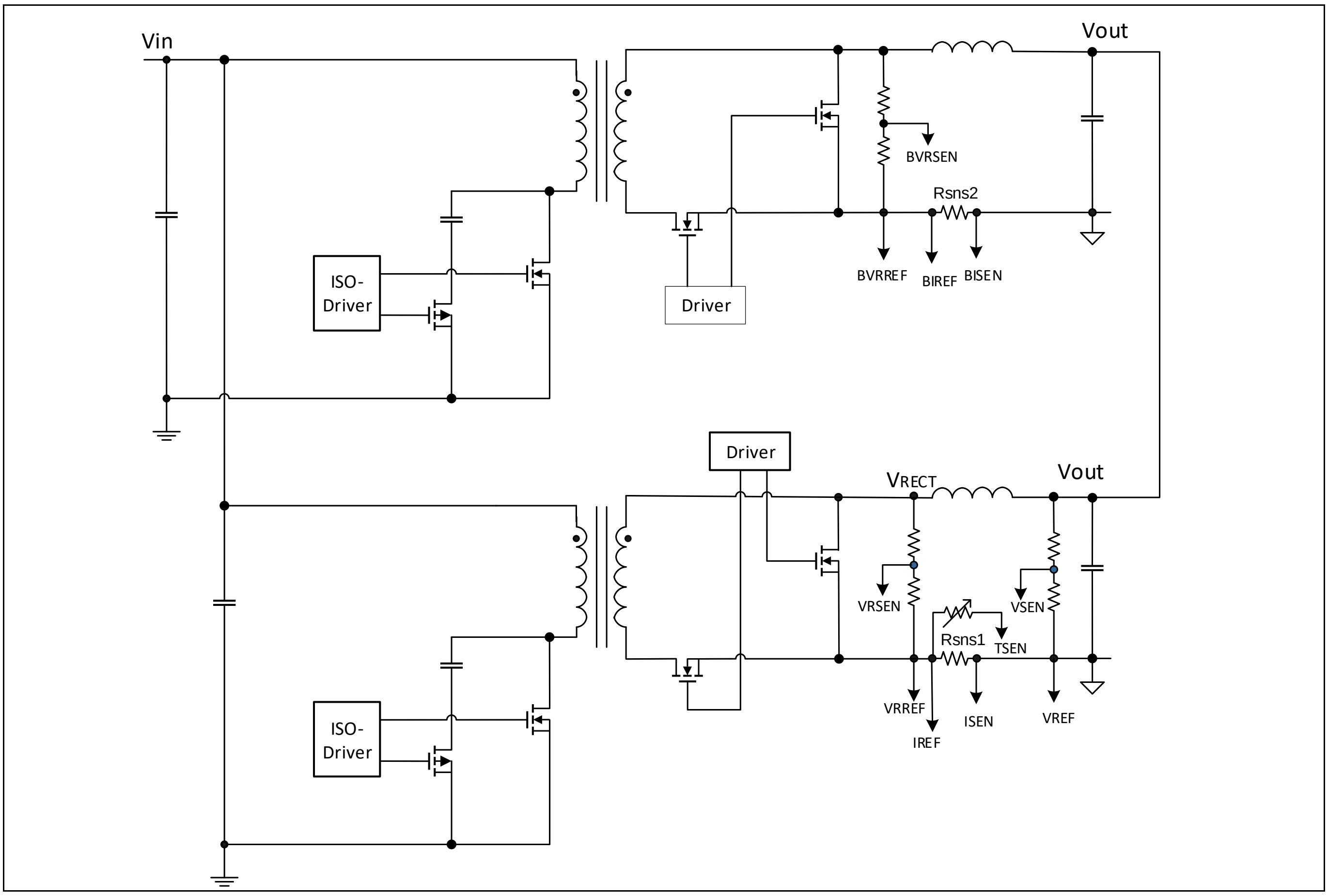


Figure 9 Interleaved ACF

1.5 TS ADC

The telemetry sense (TS) ADC is a general-purpose ADC (1 MHz, 9-bit, 0 to 1.2 V range). The general-purpose ADC consists of eight channels and can be configured to digitize voltages, currents, impedance and temperature (**Figure 10**).

- PRISEN, primary voltage sensing
- IMON, current monitoring and active current sharing
- ATSEN, temperature sense A
- BTSEN, temperature sense B
- XADDR1, address offset 1
- XADDR2, address offset 2
- ITSEN, internal temperature sense

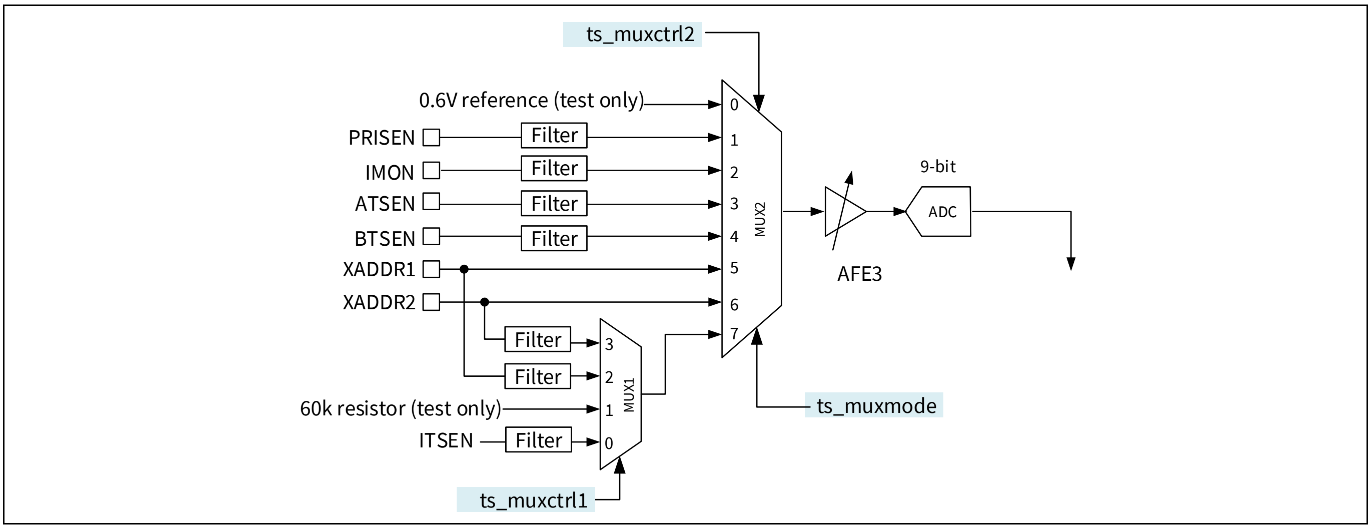


Figure 10 TS block (TS ADC)

The conversion sequence of channels can be defined by the user or can be set to auto-sequencing per MUX registers. Each input channel of the TS ADC has its own enable/disable registers. The function of each TS ADC input will be described in individual chapters in this document.

Table 1 TS ADC MUX control registers

Register name	Description
ts_muxctrl1	TS ADC MUX1 input source select. The output of MUX1 is connected to MUX2 input 7. 0 = ITSEN, internal temperature sense 1 = 60 k resistor (test only) 2 = XADDR1 filtered 3 = XADDR2 filtered
ts_muxctrl2	TS ADC MUX2 input source select. The output of MUX2 is connected to the TS ADC input. 0 = 0.6 V reference (test only) 1 = PRISEN 2 = IMON

Register name	Description
	3 = BTSEN 4 = ATSEN 5 = XADDR1 unfiltered 6 = XADDR2 unfiltered 7 = MUX1 output
ts_maxmode	TS ADC input sequence control. When bit [2] is 0, the TS ADC input is entirely determined by the settings of ts_muxctrl1 and ts_muxctrl2. When bit [2] is 1, MUX2 auto-sequences its input using the pattern in the table below. If the sequences include MUX2 input 7 (MUX1), the source in this timeslot is determined by the setting of ts_muxctrl1. 0 to 3 = defined by ts_muxctrl1, 2 4 = auto sequence: 1, 2, 1, 3, 1, 2, 1, 4 5 = auto sequence: 1, 5, 1, 3, 1, 7, 1, 4 6 = auto sequence: 1, 5, 2, 3, 1, 7, 2, 4 7 = auto sequence: 1, 5, 2, 3, 1, 7, 6, 4

1.6 PWM

The pulse width modulator converts the duty-cycle into one or more PWM pulses depending on the application. The XDPP1100 has up to 12 PWM output pins. The PWM consists of:

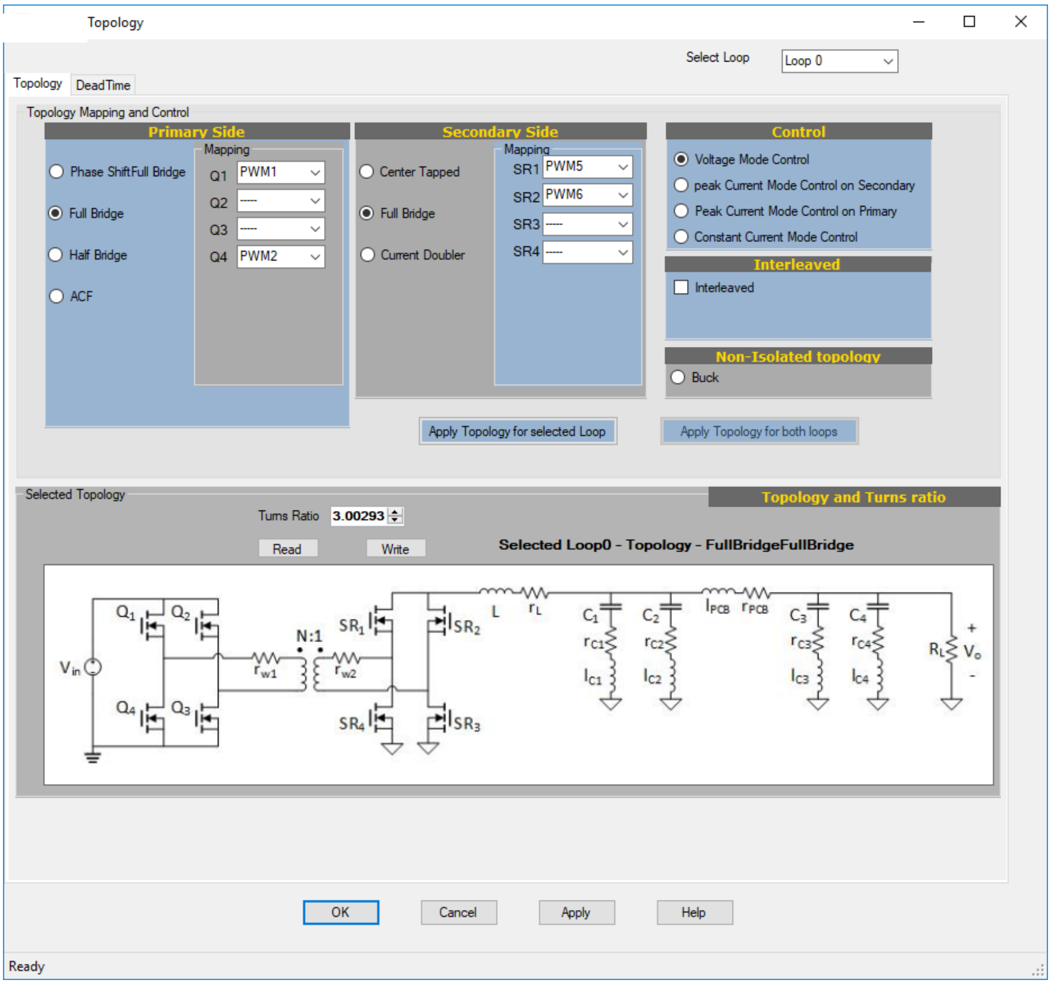
- Two ramp generators
- 12 pulse generators
- 12 interpolators

Key features of the PWM modulator:

- 78.125 ps PW resolution
- Trailing-edge, leading-edge or dual-edge modulation
- Adjustable phase shift between outputs
- PWMx re-mappable
- Cycle-by-cycle duty-cycle matching
- Adjustable dead-time between pairs for both rising and falling edges
- Dead-time resolution 1.25 ns
- Up to 2 MHz switching frequency
- Frequency/period resolution 20 ns

The XDPP1100 allows mapping PWM output freely to any primary or secondary gate drive. The XDPP1100 GUI provides a design tool to map PWM based on a schematic view.

All of the PWM outputs are also GPIO pins. Each of the GPIO pins can be configured by software as input (with or without pull-up or pull-down) or as output (push-pull or open-drain). All the GPIO pins are shared with alternative functions. The I/O configuration can be locked if necessary to follow a specific function. The GPIO configuration is programmed by FW_CONFIG_PMBUS PMBus command and is described in chapter 13.



Topology

Select Loop: Loop 0

Topology | DeadTime

Topology Mapping and Control

Primary Side

○ Phase ShiftFull Bridge
● Full Bridge
○ Half Bridge
○ ACF

Mapping
Q1 PWM1
Q2 ----
Q3 ----
Q4 PWM2

Secondary Side

○ Center Tapped
● Full Bridge
○ Current Doubler

Mapping
SR1 PWM5
SR2 PWM6
SR3 ----
SR4 ----

Control

● Voltage Mode Control
○ peak Current Mode Control on Secondary
○ Peak Current Mode Control on Primary
○ Constant Current Mode Control

Interleaved

□ Interleaved

Non-Isolated topology

○ Buck

Selected Topology

Topology and Turns ratio

Turns Ratio 3.00293

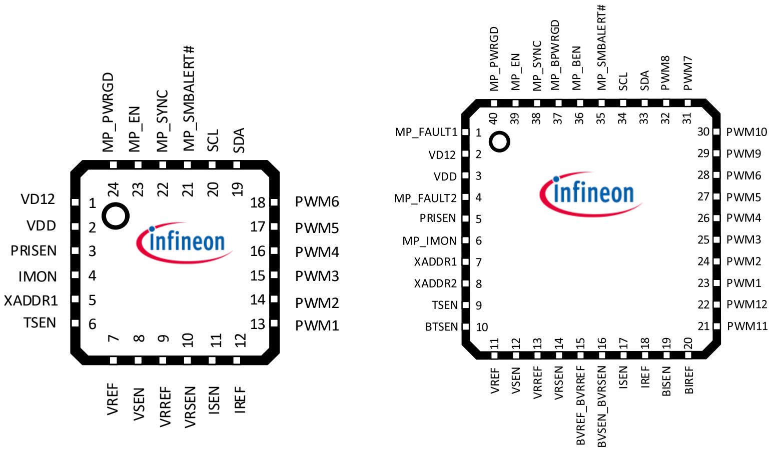
Selected Loop0 - Topology - FullBridgeFullBridge

Ready

Figure 11 GUI design tool – topology selection and PWM mapping

1.7 Pin-out

The pin-out diagrams of the three variants are shown in **Figure 12**. Please refer to the datasheet for pin definitions.



(a) XDPP1100-Q024 (b) XDPP1100-Q040

Figure 12 XDPP1100 pin-out

2 Input voltage sensing and feed-forward

2.1 How does the digital controller sense V_{IN} in an isolated converter?

Digital controllers have been used in isolated power supplies in recent years to get the benefits of high performance, flexibility, and capability of managing complex systems. Isolated power supply refers to a power converter that uses a transformer to provide electrical isolation between the input and output terminals of the converter. The digital controller is usually placed on the secondary side of the transformer to enable more accurate control of output voltage and output current, fast output fault protections, and communication with the system (i.e. PMBus) without using an I^2C isolator. The trade-offs of a secondary-side controller include requiring an auxiliary supply to power the controller, isolated drivers to drive primary MOSFETs, complicated input voltage sensing and feed-forward implementation.

Input voltage feed-forward is a feature in which the controller uses input voltage to control output without waiting for the system feedback response. It provides faster response to avoid overshoot or undershoot caused by a slower feedback loop during input voltage transient. Input voltage feed-forward involves using input and output voltage measurement to set the nominal duty-cycle for a given topology. Fast and accurate input voltage (V_{IN}) sensing is the key to high-performance feed-forward.

Traditional input voltage sensing through an isolation boundary uses an isolated error amplifier, as shown in **Figure 13**. For better feed-forward performance, the error amplifier must have wide BW for fast response, high linearity and stability overtemperature for accurate control. This approach requires additional primary and secondary bias circuits.

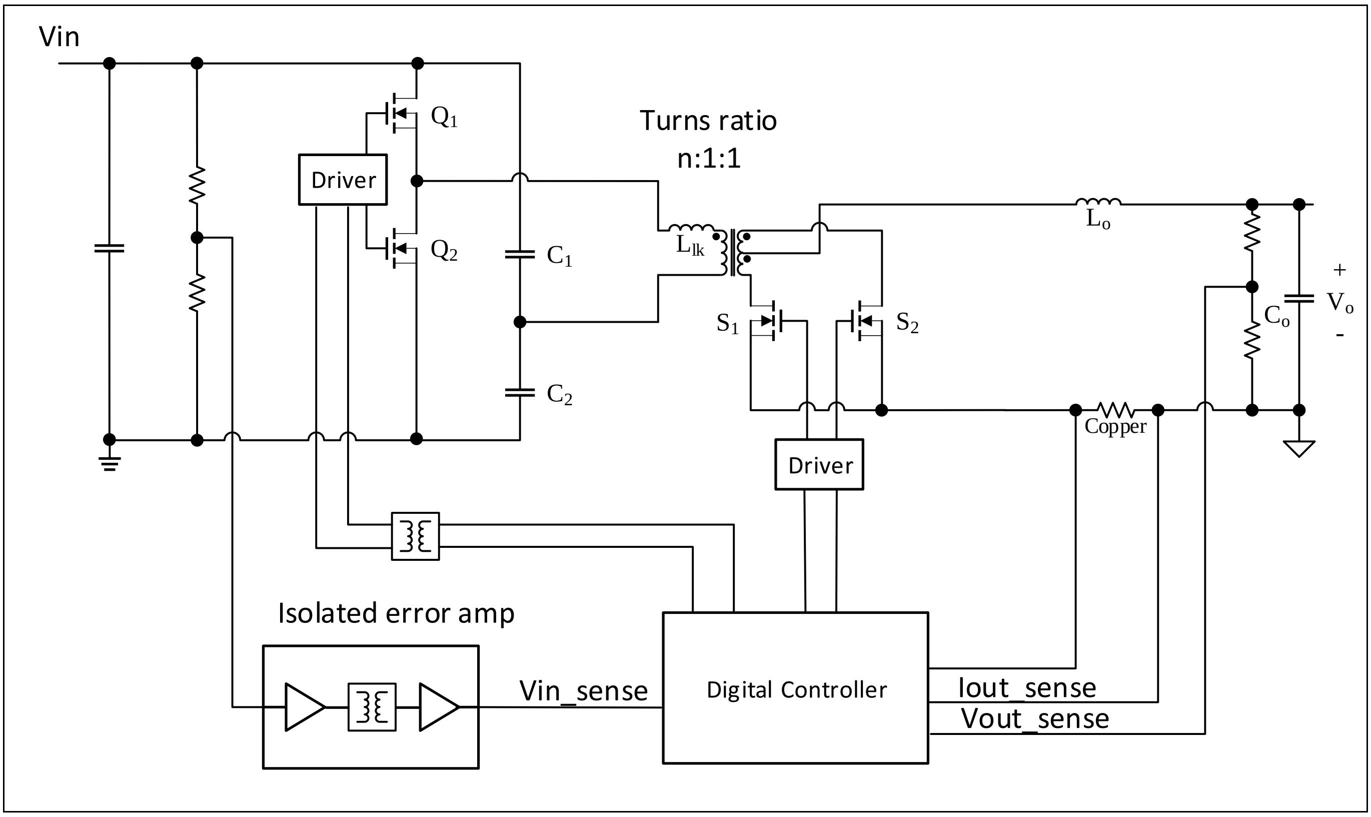


Figure 13 V_{IN} sense through isolated error amplifier

Another option for input voltage sensing is via an auxiliary transformer. The transformer must have a secondary winding with forward polarity connection so it can provide an output voltage that is proportional to the input. This auxiliary transformer could be the same housekeeping transformer that provides bias voltage for the system. The drawback of this solution is the voltage would be sensitive to the operation status – i.e. the sensed voltage varies slightly when the main converter turns on/off, as the load of the bias circuit changes with the on/off state.

Many advanced digital controllers support secondary-side input voltage sensing by measuring the rectified voltage (V_{RECT}), as shown in **Figure 14**. V_{RECT} is the transformer secondary voltage after diodes or SR MOSFET rectifiers. The amplitude of V_{RECT} voltage is proportional to the input voltage V_{IN} by the scale of the transformer turns ratio. The V_{RECT} voltage could be further scaled down by a resistor divider and fed to the input of a digital controller.

Compared to isolated error amplifier or auxiliary transformer solutions, V_{RECT} sensing eliminates additional components (isolated error amplifier, its bias circuit and external components, or transformer and its rectifier circuit and filter), reduces the bill of materials (BOM), reduces layout footprint, reduces power loss and increases system reliability. The downside of the V_{RECT} sensing is that the V_{RECT} signal is not available when the main converter stops switching. Thus, it cannot measure the input voltage in off mode.

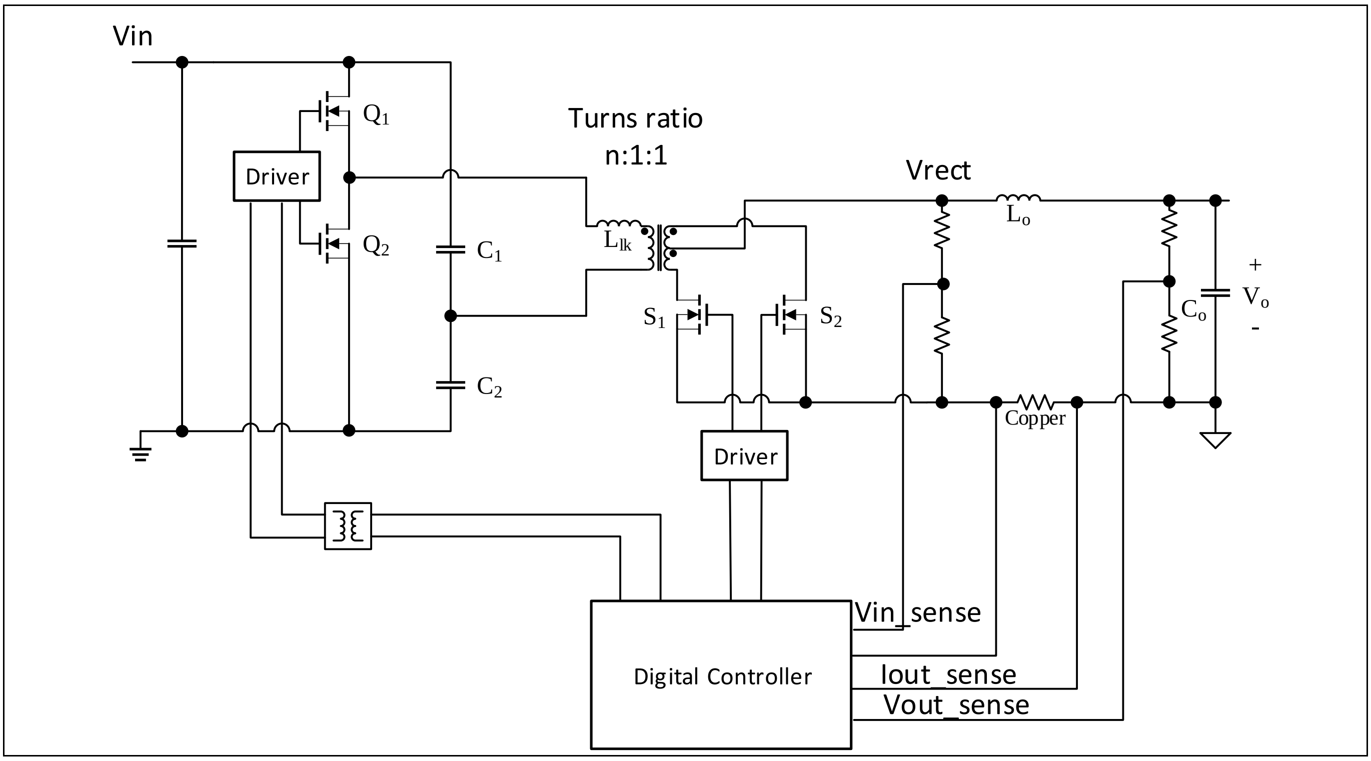


Figure 14 V_{IN} sense by V_{RECT} sensing

The XDPP1100 supports all these input voltage sensing methods using two types of voltage ADCs. The VADCs (VSEN, VRSEN and BVSEN_BVRSEN) have much higher performance than the TS ADC (PRISEN) – see Table 2 for the comparison. The user could select the input voltage sensing method based on system requirements. For example, a FB converter with voltage mode control (VMC) would prefer to use the VRSEN in V_{RECT} sense mode. This setup provides accurate input voltage sensing, fast feed-forward response and flux balancing feature.

Table 2 VSEN_VRSEN vs. PRISEN

	ADC	Sample rate (Msps)	Resolution (mV)	Range (V)
VSEN/VRSEN/BVRSEN (VADC)	11-bit	100	1.25	0 to 2.1
PRISEN (TS ADC)	9-bit	1	2.344	0 to 1.2

2.2 Input voltage source

The XDPP1100 offers a wide range of source selection for input voltage sensing. Table 3 lists the input voltage sources supported by the XDPP1100. Options 0, 1, 3, 5, 6 and 7 are all good for isolated converters. VRSEN and BVSEN_BVRSEN are recommended for the input voltage sensing, because the VADC can achieve accurate and fast feed-forward response.

Table 3 tlm_vin_src_sel configuration table

Value	Input voltage source
0	VRSEN. Secondary V_{RECT} sense, vrs_init prior to start-up
1	BVSEN_BVRSEN. Secondary V_{RECT} sense, vrs_init prior to start-up
2	Loop 0 V_{OUT}. Select on loop 1 when loop 1 V_{IN} provided by loop 0 V_{OUT} (e.g., post-buck)
3	TS ADC V_{IN}. non-pulsed/primary V_{IN} sense via telemetry ADC (PRISEN)
4	tlm_vin_force. Forced V_{IN} via FW (e.g., FW override of HW computation)
5	VRSEN. Secondary V_{RECT} sense, 0 V prior to start-up. Select on loop 1 when sharing loop 0 V_{RECT} sense
6	VRSEN. Non-pulsed/primary V_{IN} sense
7	BVSEN_BVRSEN. Non-pulsed/primary V_{IN} sense

Options 0, 1, and 5 use VRSEN or BVSEN_BVRSEN in the V_{RECT} sense mode (**Figure 14**). Since the V_{RECT} doesn't know the input voltage prior to start-up, it is necessary to use an initial voltage vrs_init to initialize the first pulses (options 0 and 1). The vrs_init is configured by the **vrs_voltage_init** register. For the second loop (loop 1), if it shares loop 0 V_{RECT} sense, it can only be enabled after loop 0 starts regulation. Thus, the initial voltage of loop 1 is not critical and could be set to 0 V (option 5).

Options 6 and 7 use VRSEN or BVSEN_BVRSEN in non-pulsed or DC mode. This configuration could be selected when the input voltage is sensed through an isolated op-amp or auxiliary transformer, as described earlier. The DC mode of VRSEN or BVSEN_BVRSEN is configured by register **vsp1_vrs_sel** (for VRSEN) or **vsp2_vrs_sel** (for BVSEN_BVRSEN).

Option 3 uses PRISEN input to sense non-pulsed input signal. Since the TS ADC is slower than VADC, the PRISEN should only be used when the VRSEN is not available for DC input sensing. An example is shown in chapter 2.8.

2.3 XDPP1100 V_{RECT} sensing

Figure 15 is a typical isolated FB converter with center-tap (CT) rectifier using the XDPP1100-Q024. The VSEN is used for output voltage V_{OUT} sensing, and the VRSEN is used for V_{RECT} voltage sensing. The differential input sense enables routing the sense and reference voltage in pairs to minimize common mode noise.

The input voltage V_{IN} is calculated based on the VRSEN voltage, the V_{RECT} resistor divider ratio, and the transformer turns ratio. There are two manufacturer-specified PMBus commands for the user to set up the ratio. The MFR_VRECT_SCALE is the resistor divider ratio of V_{RECT} (**Figure 19**). The MFR_TRANSFORMER_SCALE defines the transformer turns ratio. This transformer scale is set per N_S/N_P.

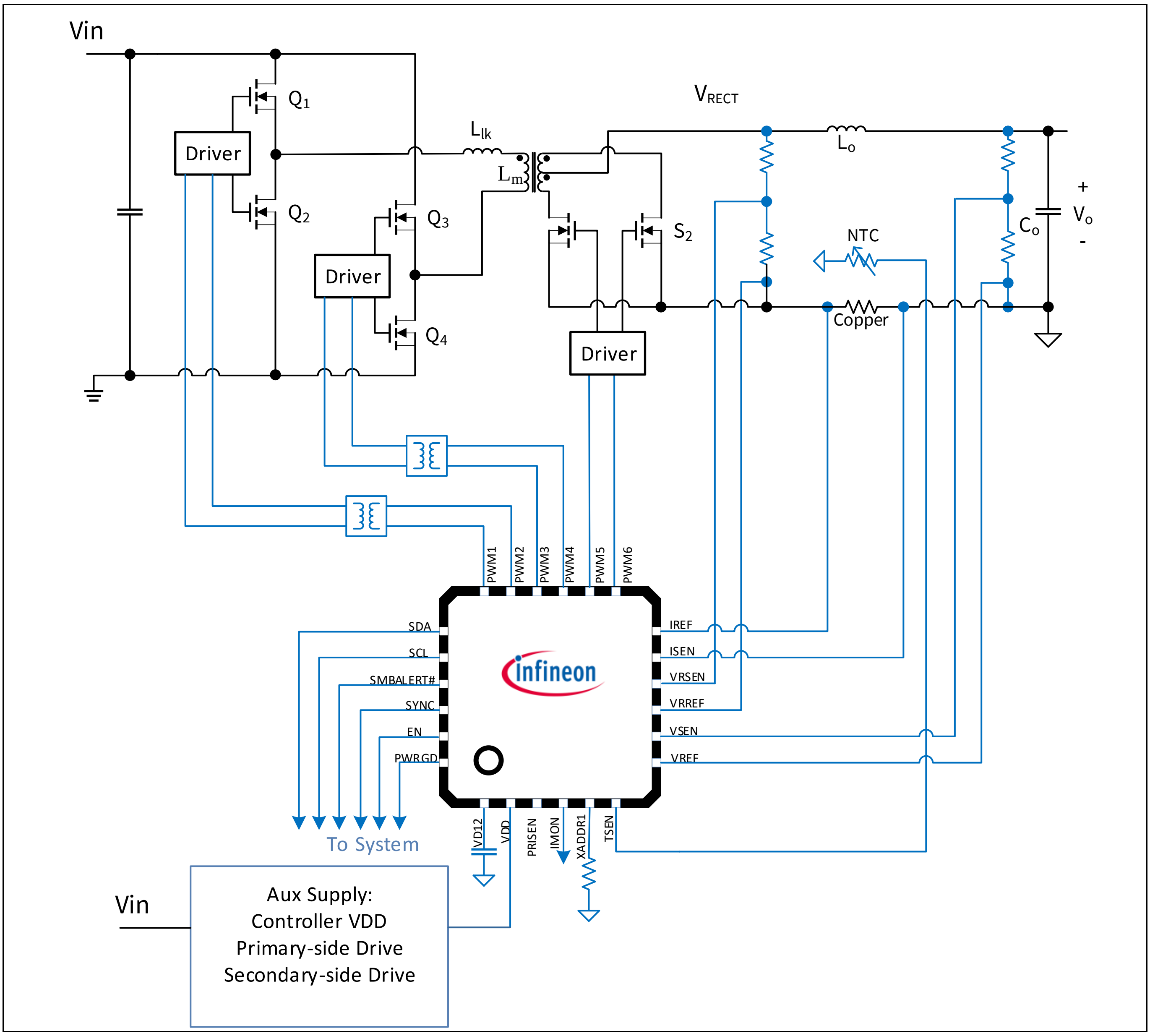


Figure 15 XDPP1100 FBCT VMC

The VRSEN ADC can be configured to be DC mode or VRS mode by register **vsp1_vrs_sel**. When VRS mode is configured to measure the rectified voltage waveform (V_{RECT}), it enables the rectified voltage sense processor and VRS edge comparator. **Figure 16** waveforms illustrate how the sampling of the V_{RECT} waveform works. VRSEN is the scaled V_{RECT} signal. Noise has been added to the ideal VRSEN waveform to highlight the importance of sampling window timing.

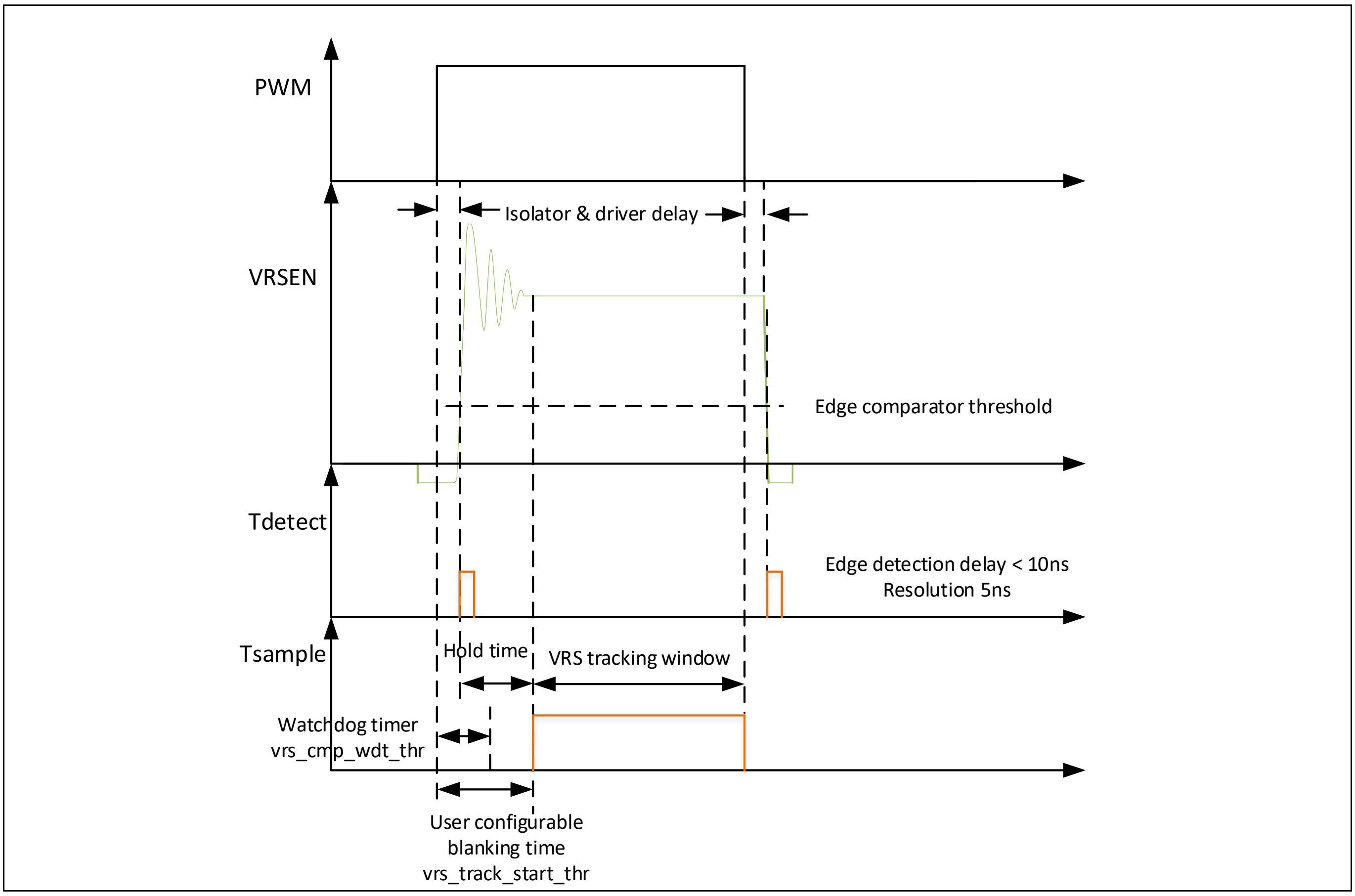


Figure 16 **Timing of V_{RECT} sensing by VRSEN**

2.3.1 VRS tracking window

The VRS edge detector works at 200 MHz clock. It senses the rising and falling edges of VRSEN waveforms with delay less than 10 ns. The T_{detect} waveform is the output of edge detector logic. The rising edge of the rectified voltage waveform is detected, and a programmable hold time and blanking window is added to it. The hold time should be set long enough for the tracking ADC to settle (more than 250 ns). The blanking time should be configured longer than the voltage spike and ringing duration. **vrs_track_start_thr** is the register that defines V_{RECT} tracking start time.

A watchdog timer (**vrs_cmp_wdt_thr**) is added to the VRS comparator to monitor the quality of the V_{RECT} signal. It measures from the rising edge of the PWM. If VRSEN has not tripped the edge comparator by the time the time-out threshold is reached, it is assumed VREN is below the comparator threshold and the V_{RECT} sense will enter a hold phase. This watchdog timer should be set to a value greater than the expected time for V_{RECT} to go high, i.e. longer than the combined isolator and gate driver delays. It also needs to be set less than the tracking start threshold defined by **vrs_track_start_thr**.

The hold phase starts when one of the conditions is met: either by edge detection at the rising edge of VRSEN, or by the watchdog timer **vrs_cmp_wdt_thr**. The hold phase ends when the **vrs_track_start_thr** timer is completed, then the ADC becomes active and starts tracking voltage. The hold time should be set longer than 250 ns. i.e. **vrs_track_start_thr** - **vrs_cmp_wdt_thr** should be longer than 250 ns. LSB of these two registers is 10 ns; i.e., writing 30 to **vrs_track_start_thr** sets the blanking time to 300 ns.

The edge comparator has two configurable reference voltage thresholds: 500 mV and 300 mV, defined by register **vrs_cmp_ref_sel**. The user can choose a proper threshold based on the VRSEN signal level. To optimize VRSEN accuracy, it is recommended to scale V_{RECT} voltage to VRSEN in the 600 mV to 2.1 V range. For example, a telecom brick application has input voltage 36 V to 75 V. The V_{IN} undervoltage (UV) fault threshold is 30 V, V_{IN} overvoltage (OV) fault threshold is 80 V; transformer turns ratio is 3:1. V_{RECT} amplitude spans from 10 V to 27 V between the two V_{IN} fault limits. Setting the V_{RECT} resistor divider ratio to 0.07 scales V_{RECT} to 0.7 V~1.89 V, which nicely fits into the VRSEN input voltage range.

After the hold and blanking window, sampling of the rectified voltage can occur (shown as the T_{sample} waveform). The sampling window ends when the associated PWM signal goes low. If input voltage changes during this period, VRSEN ADC tracks the change. At the end of a sampling window, VRSEN ADC remembers the value of the last ADC sample and uses this value for the feed-forward computation for the next switching cycle.

If the PWM and V_{RECT} pulse widths are very narrow, which happens during start-up, and are shorter than the blanking time for a valid measurement, the VRSEN ADC will stay in hold mode until the V_{RECT} pulse is long enough for effective tracking and measurement. In start-up, VRSEN ADC typically holds the preset initial voltage in a fresh start-up or holds the last measurement at the previous shutdown during a restart.

The VRSEN ADC is preset at the value of the last cycle to minimize the time required for the ADC to properly track the voltage.

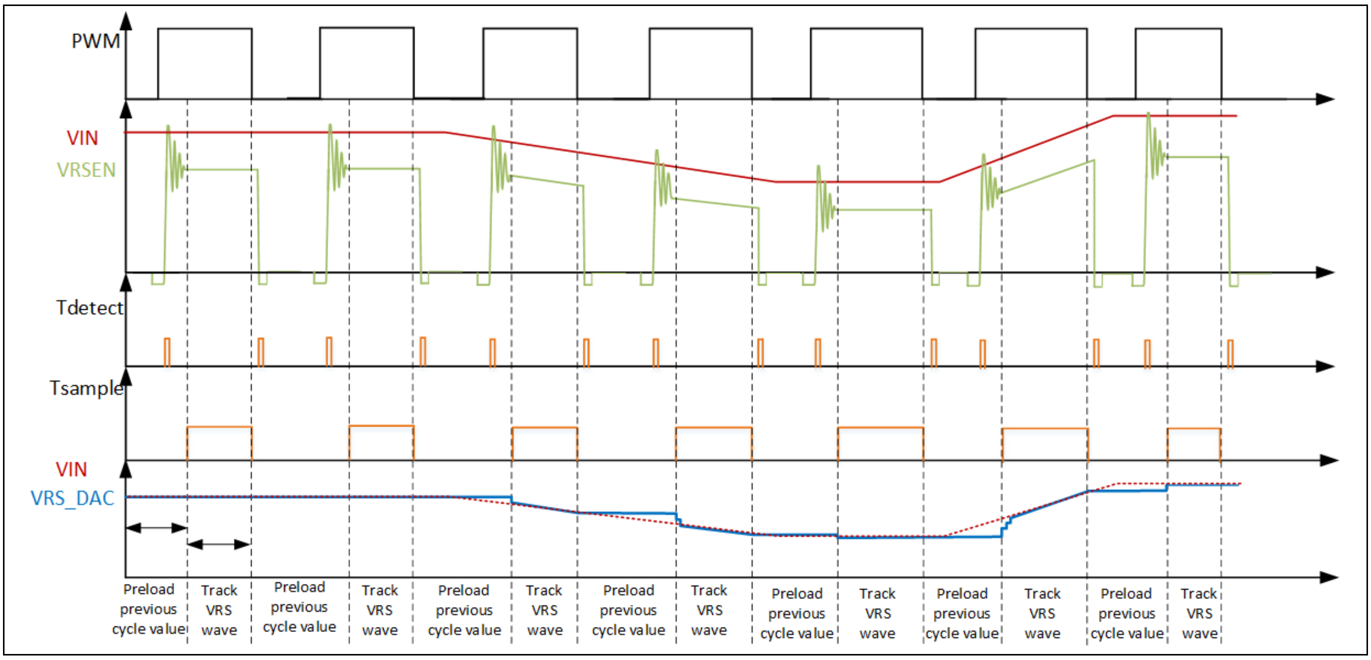

PWM
VIN
VRSEN
Tdetect
Tsample
VIN
VRS_DAC
Preload previous cycle value
Track VRS wave
Preload previous cycle value
Track VRS wave
Preload previous cycle value
Track VRS wave
Preload previous cycle value
Track VRS wave
Preload previous cycle value
Track VRS wave
Preload previous cycle value
Track VRS wave
Preload previous cycle value
Track VRS wave

Figure 17 V_{RECT} **sensing during input transient**

2.3.2 VRS initial voltage

At start-up, the XDPP1100 uses a user-configurable initial input voltage (**vrs_voltage_init**) to enable a converter that allows the XDPP1100 to not remain stuck in unknown input status. The initial input voltage is typically set to nominal input voltage per application. During start-up, PWM pulse width increases during the user-defined rise time. As soon as the PWM pulse is long enough for valid V_{RECT} measurement, the input voltage telemetry will transition from the initial voltage to actual measured voltage (**Figure 18**). Given this behavior, V_{RECT} based input voltage sensing is not able to detect V_{IN} OV or UV status prior to switching actually starting.

Register **vrs_voltage_init** should be calculated based on topology.

$$vrs\_voltage\_init = \frac{Vin\_init(V)}{20mV} \cdot MFR\_VRECT\_SCALE \cdot MFR\_TRANSFORMER\_SCALE$$, for FB or ACF

Input voltage sensing and feed-forward

$$vrs\_voltage\_init = \frac{Vin\_init(V)/2}{20mV} \cdot MFR\_VRECT\_SCALE \cdot MFR\_TRANSFORMER\_SCALE \text{ , for HB topology}$$

Here *Vin_init* is initial input voltage, and the MFR_VRECT_SCALE is the resistor divider ratio of V_{RECT} (**Figure 19**).

MFR_TRANSFORMER_SCALE defines the transformer turns ratio. This scale is set per N_S/N_P.

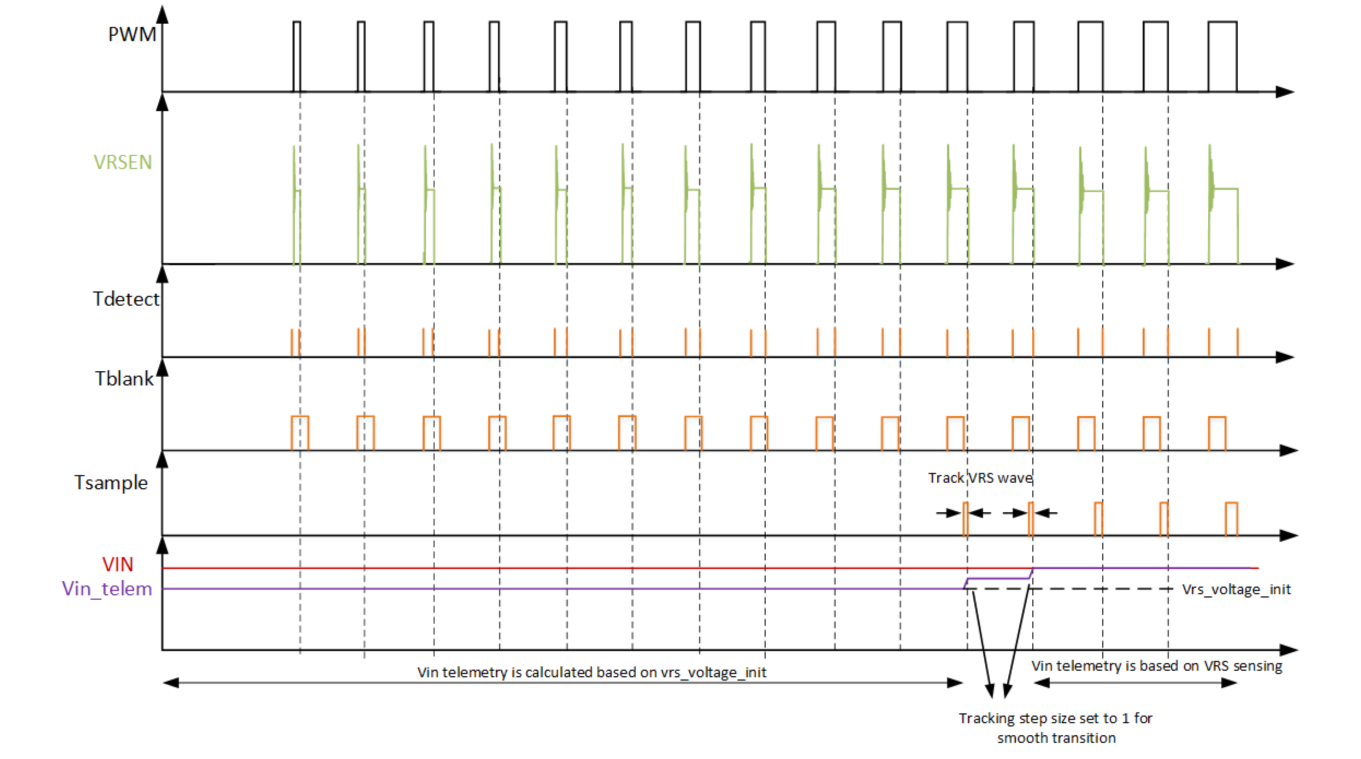
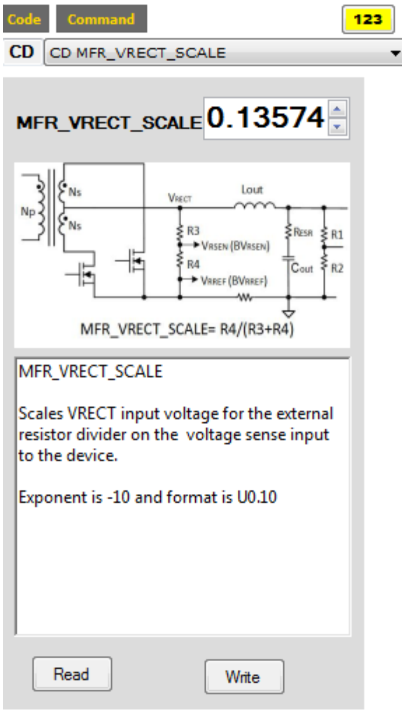

PWM
VRSEN
Tdetect
Tblank
Tsample
Track VRS wave
VIN
Vin_telem
Vrs_voltage_init
Vin telemetry is calculated based on vrs_voltage_init
Vin telemetry is based on VRS sensing
Tracking step size set to 1 for smooth transition

Figure 18 V_{RECT} **sensing during start-up**



MFR_VRECT_SCALE 0.13574

MFR_VRECT_SCALE= R4/(R3+R4)

MFR_VRECT_SCALE

Scales VRECT input voltage for the external resistor divider on the voltage sense input to the device.

Exponent is -10 and format is U0.10

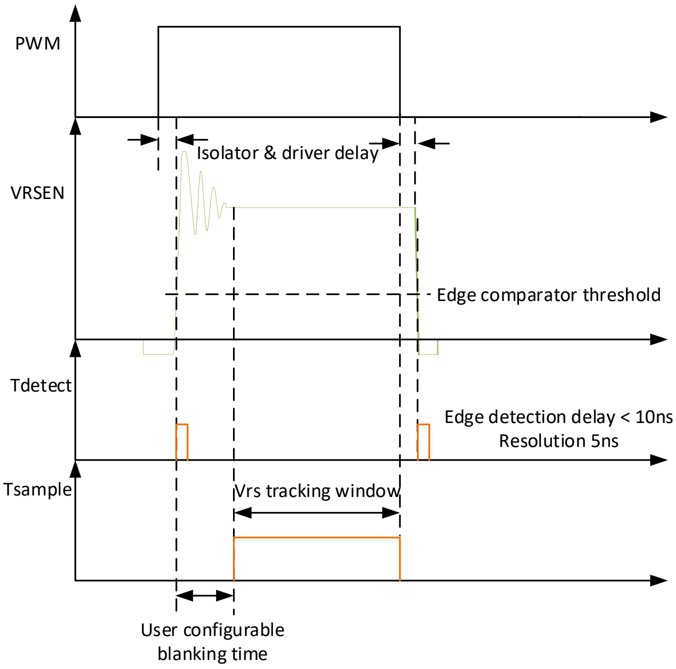

PWM
VRSEN
Tdetect
Tsample
Isolator & driver delay
Edge comparator threshold
Edge detection delay < 10ns
Resolution 5ns
Vrs tracking window
User configurable blanking time

Figure 19 **MFR_VRECT_SCALE**

2.3.3 V_{RECT} sensing compensation

To improve the accuracy of input voltage telemetry, the XDPP1100 offers HW-based compensation.

The voltage drop on transformer DC resistance (DCR), copper trace DCR, and $R_{DS(on)}$ of SR MOSFETs could be compensated by register **tlm_vrect_rcorr**. This register defines the total equivalent DC resistance of the V_{RECT} sensing loop that reflects to primary. The V_{IN} telemetry is compensated by **tlm_vrect_rcorr** * I_{OUT}.

The voltage offset could be compensated by register **tlm_vrect_offset**. The V_{IN} telemetry is compensated by **tlm_vrect_offset**/MFR_VRECT_SCALE/MFR_TRANSFORMER_SCALE.

The XDPP1100 also considers the body diode voltage drop when the SR MOSFETs are turned off, i.e. in DE mode. The voltage drop is configured by register **tlm_vrect_sr_diode**, with the choice of 0 mV, 480 mV, 640 mV, 800 mV, 960 mV, 1280 mV, 1600 mV and 1920 mV.

Please see Table 10 for the description of each register.

2.3.4 VRS de-glitch

When the converter operates in hard-switching mode, the V_{RECT} node has more noise on the rising and falling edge. In some cases the noise rings up and down and crosses the comparator threshold multiple times during the rising edge, as shown in **Figure 20** (blue trace). To avoid the noise tripping the VRS edge comparator more than once, a digital de-glitch is added. **vrs_min_pw** sets the VRS de-glitch pulse width. Setting the de-glitch pulse wider than the ringing pulse width could avoid false triggering. **vrs_min_pw** adds delay to the edge detection. The delay is not so critical for input voltage sensing and feed-forward but would affect volt-second flux balancing performance for FB topologies. Thus, a good PCB layout is always the first priority to avoid needing to use the de-glitch filter. See chapter 14 for the XDPP1100 layout guidelines.

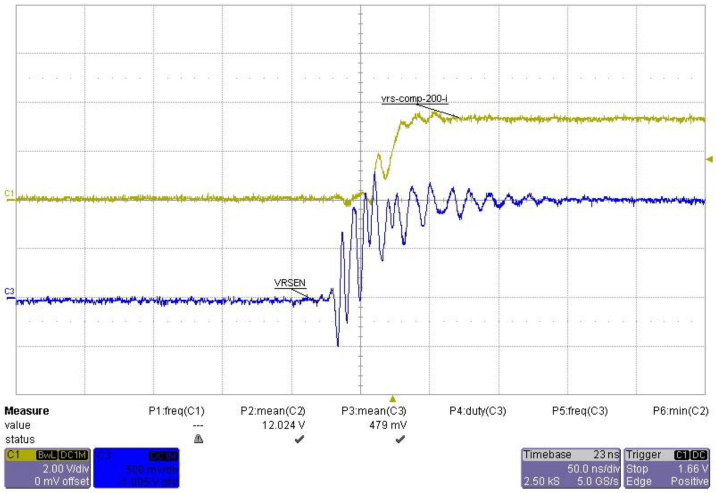


Figure 20 VRSEN noise

The XDPP1100 GUI offers a design tool to help the user configure the VRS sensing registers. See chapter 2.7.

2.4 XDPP1100 feed-forward

The XDPP1100 computes feed-forward duty-cycle based on input voltage and output voltage. The result is added to the feedback loop PID filter output to resolve PWM duty-cycle (**Figure 21**).

$$computed\_feed\_forward = \frac{Vout,target}{Vin \times trans\_scale\_loop} \tag{2.1}$$

Here, trans_scale_loop is the transformer turns ratio, defined by N_S/N_P in FB or ACF topologies, and $N_S/(2N_P)$ in a HB topology, wherein N_P is the transformer primary turns number, and N_S is the transformer secondary turns number. The user can define the transformer scale by PMBus command MFR_TRANSFORMER_SCALE (N_S/N_P) for all isolated topologies. The XDPP1100 FW will calculate the trans_scale_loop based on MFR_TRANSFORMER_SCALE command and device topology. FW takes care of factor 2 for the HB topology, and the user should not be concerned with setting it manually.

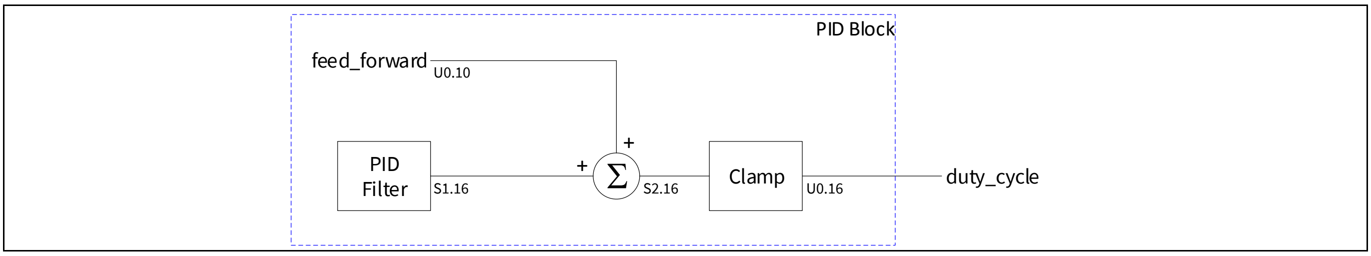


Figure 21 Duty-cycle composed of the sum of the PID output and the feed-forward term

The feed-forward computation is implemented in HW and so offers the fastest response. The computation of feed-forward duty-cycle is shown in **Figure 22**.

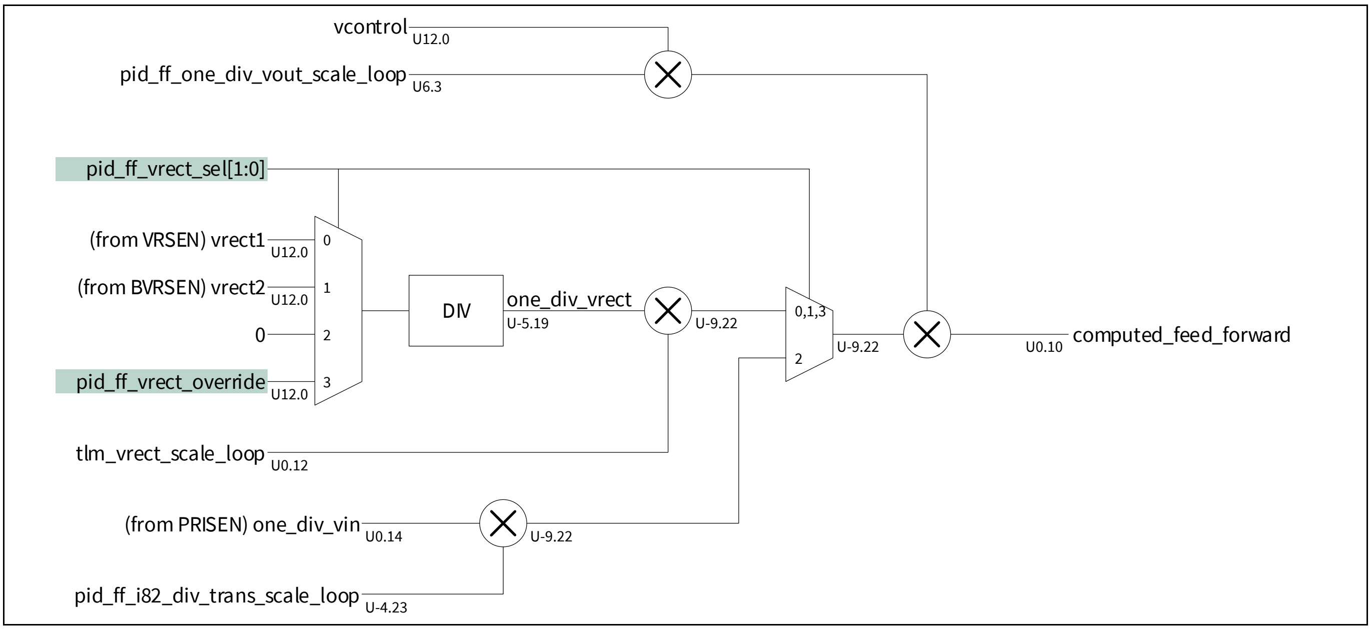


Figure 22 XDPP1100 feed-forward computation

The highlighted signals in **Figure 22** are accessible through the register map. A detailed description of each register can be found in chapter 2.5.

The product vcontrl * pid_ff_one_div_vout_scale_loop is clamped to U16.0. Thus, to use the HW-based feed-forward, the maximum output voltage is limited to (2^16-1) * 1.25 mV = 81.92 V. Here 1.25 mV is the LSB of the voltage sense ADC. This limitation doesn't apply if using FW-computed feed-forward override (**Figure 25**).

For buck topology without a transformer, the transformer scale should be set to the maximum (0.999). For other non-isolated topologies such as boost and buck-boost, the feed-forward computation is different and will be discussed in a separate document.

Please note that the feed-forward function only applies to VMC, and is not applicable to PCMC.

2.4.1 Feed-forward V_{OUT} voltage calculation

The XDPP1100 uses the $V_{control}$ voltage to calculate V_{OUT} for feed-forward computation. $V_{control}$ is an internal reference voltage that controls the target output voltage. $V_{control}$ is equal to $V_{OUT,\ target}$ multiplied by resistor divider VOUT_SCALE_LOOP (**Figure 23**). VOUT_SCALE_LOOP is a PMBus command.

$$V_{out,target} = \frac{v_{control}}{vout\_scale\_loop} \qquad (2.2)$$

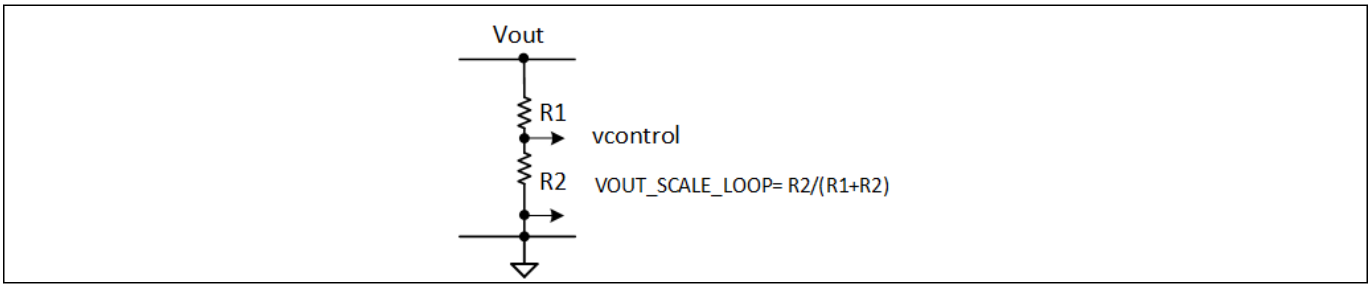


Figure 23 VOUT_SCALE_LOOP definition

2.4.2 Feed-forward V_{IN} source selection

The input voltage used for feed-forward computation should be selected from the following source, by register **pid_ff_vrect_sel**. It is worth mentioning that the V_{IN} telemetry source select and the V_{IN} feed-forward source select are independent. The user has the flexibility to use the same input sense pin for V_{IN} telemetry and feed-forward or use a different input source. For example, one use case is using PRISEN for input voltage telemetry and using the VRSEN for feed-forward and flux balancing.

Table 4 pid_ff_vrect_sel configuration table

0	VS1 (VRSEN) Vrect1
1	VS2 (BVSEN_BVRSEN) Vrect2
2	TS V_{IN} (PRISEN)
3	pid_ff_vrect_override

Here Vrect1 and Vrect2 are the input voltage of the VRSEN or BVRSEN pins. Typically, it is transformer secondary rectifier voltage scaled down by V_{RECT} resistor divider (**Figure 24**) in an isolated converter.

$$VRSEN = Vrect \times vrect\_scale\_loop \qquad (2.3)$$

V_{RECT} voltage is proportional to V_{IN}:

$$Vrect = Vin \times trans\_scale\_loop \qquad (2.4)$$

Then V_{IN} could be calculated by vrect_scale_loop, trans_scale_loop and the VRSEN voltage.

For non-isolated applications, the V_{IN} could be directly sensed through a resistor divider. Even though no V_{RECT} is sensed in this case, the input voltage used for the feed-forward computation should still be configured by the **pid_ff_vrect_sel** register. Don’t be confused by the name of the register. The example of the non-isolated or DC mode VRSEN is demonstrated in chapter 2.4.4.2.

The XDPP1100 offers an option for FW to override the feed-forward input voltage. If an override input voltage is desired, set the **pid_ff_vrect_sel** =3 and calculate **pid_ff_vrect_override** with:

$$pid\_ff\_vrect\_override\ = \frac{Vin(V) \times trans\_scale\_loop \times rect\_scale\_loop}{1.25(mV)} \qquad (2.5)$$

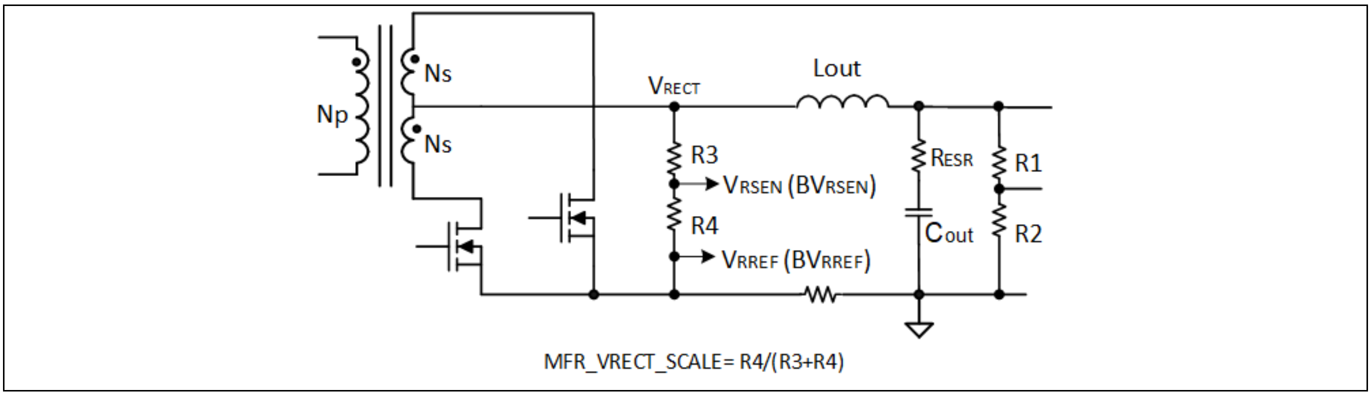


Figure 24 MFR_VRECT_SCALE definition

Depending on which input voltage source is selected, the computed feed-forward duty-cycle is calculated:

Table 5 computed_feed_forward table

pid_ff_vrect_sel	computed_feed_forward
0	$computed\_feed\_forward = \frac{vcontrol}{vout\_scale\_loop} \times \frac{1}{VRSEN} \times vrect\_scale\_loop$
1	$computed\_feed\_forward = \frac{vcontrol}{vout\_scale\_loop} \times \frac{1}{BVRSEN} \times vrect\_scale\_loop$
2	$computed\_feed\_forward = \frac{vcontrol}{vout\_scale\_loop} \times \frac{1}{Vin\_by\_PRISEN \times trans\_scale\_loop}$
3	$computed\_feed\_forward = \frac{vcontrol}{vout\_scale\_loop} \times \frac{1}{pid\_ff\_vrect\_override} \times vrect\_scale\_loop$

The TS PRISEN is one of the input channels of the TS ADC. The conversion sequence of channels can be defined by the user, or can be set to auto-sequencing per MUX registers. See chapter 2.7.1.2 for details.

2.4.3 Feed-forward override and adjustment options

2.4.3.1 Feed-forward override

The XDPP1100 allows overriding the feed-forward duty-cycle with a user-defined **pid_ff_override** value, selected by register **pid_ff_override_sel** (**Figure 25**). This feature gives an option for the FW to completely override the computed feed-forward value. **pid_ff_override** register should be calculated by:

$$pid\_ff\_override = ff\_override\_duty\_cycle \times 2^{10} \qquad (2.6)$$

For example, if the user wishes to override feed-forward duty-cycle with a value of 62.5 percent, **pid_ff_override** register should set to 0.625 * 1024 = 640.

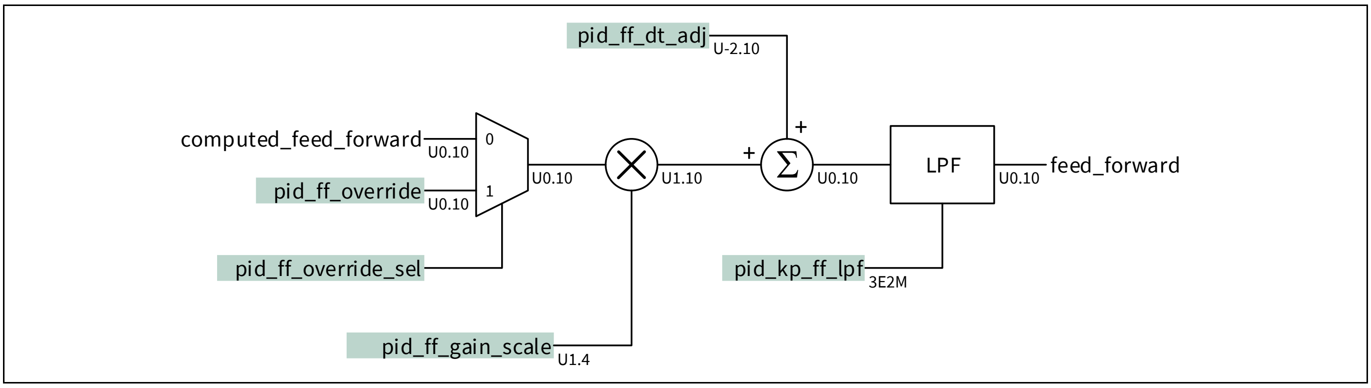


Figure 25 Feed-forward override and adjustment

2.4.3.2 Feed-forward gain and offset adjustment

The gain of feed-forward can be adjusted by **pid_ff_gain_scale**. The range of this register is 0 to 31. Setting it to 16 gives a gain of 1.0. Setting it to 0 will disable the feed-forward factor.

Register **pid_ff_dt_adj** provides feed-forward adjustment for dead-time. The computed duty-cycle discussed here is the ideal duty-cycle. The actual PWM duty-cycle is smaller than the ideal duty-cycle, because dead-time will be subtracted from the pulse width. For example, a HB converter switching at a 250 kHz frequency, at 50 percent duty-cycle with 100 ns dead-time on both high-side and low-side pulse, has 45 percent actual duty-cycle. The dead-time error of the feed-forward circuit could be compensated by the **pid_ff_dt_adj** register; i.e. if the dead-time introduces 5 percent duty-cycle error, adding 5 percent pid_ff_dt_adj will compensate for the dead-time error. Register pid_ff_dt_adj has resolution 2^{-10} and could adjust offset up to 25 percent duty-cycle.

$$pid\_ff\_dt\_adj = duty\_cycle\_adj \times 2^{10} \quad (2.7)$$

For example, to add 5 percent duty-cycle adjustment, **pid_ff_dt_adj** = 0.05 x 2^{10} = 51.

In closed-loop regulation, the dead-time error is automatically compensated by the feedback loop, so leaving the **pid_ff_dt_adj** unconfigured won't affect feed-forward performance.

2.4.3.3 Feed-forward LPF

Register **pid_kp_ff_lpf** defines the feed-forward low-pass filter (LPF). The format of **pid_kp_ff_lpf** is:

- kp_exp = pid_ff_kp_lpf [4:2]
- kp_man = 4 + pid_ff_kp_lpf [1:0]
- kp = kp_man * 2^(kp_exp – 9)
- F_{3db}(MHz) = [kp/(1 – kp)] * 50 MHz/2π

Range = 62.7 kHz (0_D) to 23.87 MHz (26_D). A handy tool in the XDP1100 GUI calculates the cut-off frequency based on register value. Please see **Figure 31**.

2.4.4 Advanced feed-forward

2.4.4.1 Same-cycle feed-forward

One limitation of V_{RECT} input voltage sensing is that the V_{RECT} can only be measured during PWM on-time. If the input voltage transient happens at a very fast slew rate, i.e. completed within one switching cycle, the V_{RECT} sensing feed-forward is not able to respond immediately and will have one switching cycle of delay. Improved input transient response could be achieved with the same-cycle feed-forward configuration. Same-cycle feed-forward enables the feed-forward duty-cycle to be calculated in real time in the same cycle of VRS sensing. Set the **vrs_same_cycle_en** register to 1 to enable VRS same-cycle mode. A go-live timing threshold is defined by **vrs_meas_start_th** (**Figure 26**). This register sets the time from the start of VRS tracking to the start of live sample updates. The test result of same-cycle feed-forward is shown in chapter 2.8.5.

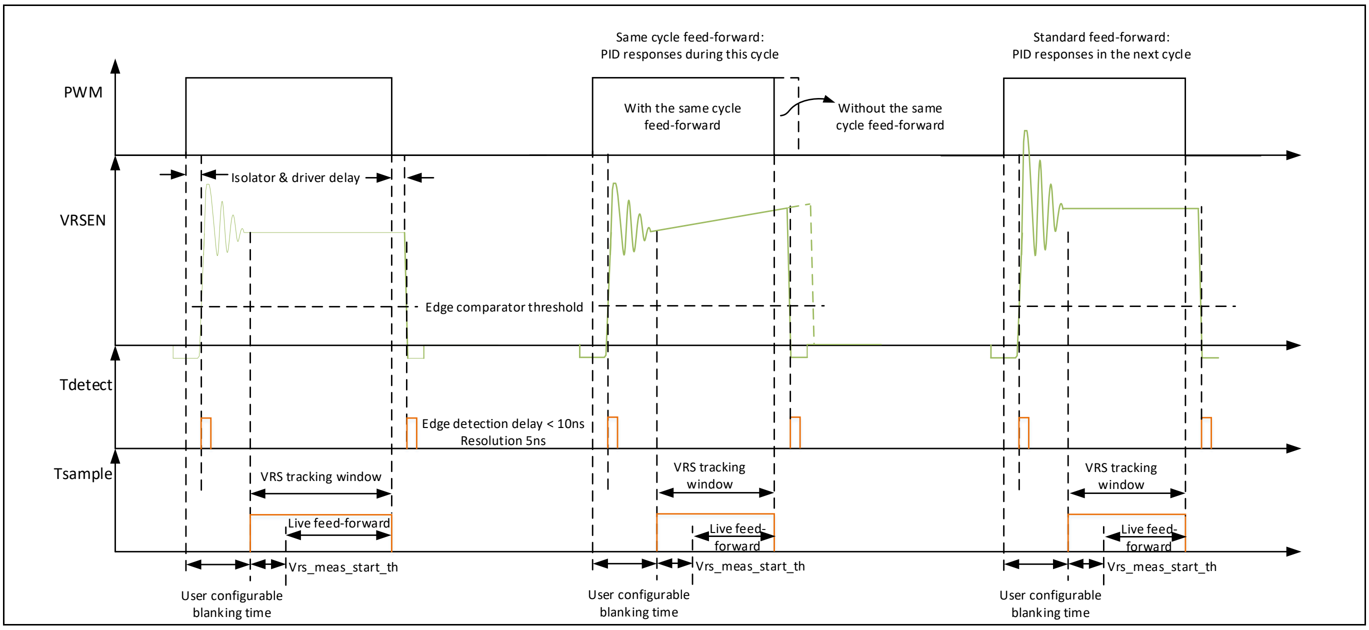

Same cycle feed-forward:
PID responses during this cycle
Standard feed-forward:
PID responses in the next cycle
PWM
With the same cycle feed-forward
Without the same cycle feed-forward
VRSEN
Isolator & driver delay
Edge comparator threshold
Tdetect
Edge detection delay < 10ns
Resolution 5ns
Tsample
VRS tracking window
VRS tracking window
VRS tracking window
Live feed-forward
Live feed-forward
Live feed-forward
Vrs_meas_start_th
Vrs_meas_start_th
Vrs_meas_start_th
User configurable blanking time
User configurable blanking time
User configurable blanking time

Figure 26 **VRS timing of the same-cycle feed-forward**

Table 6 **Same-cycle feed-forward registers**

Register name	Description
vrs_same_cycle_en	VRS same-cycle mode enabled 0 = same-cycle mode disabled 1 = same-cycle mode enabled
vrs_meas_start_th	VRS same-cycle sample counts before going to live updates LSB = 4 samples, range = 0 to 124 samples • Example: – Vrs_meas_start_th = 0, go-live updates as soon as VRS tracking starts – Vrs_meas_start_th = 1, go-live updates after 4 samples (80 ns) after VRS tracking starts

2.4.4.2 DC mode of VRS sense

The XDPP1100 also offers a DC mode operation of VRS sense. It can be used to sense input voltage like the traditional primary V_{IN} sense through an isolated error amplifier, or to directly sense input voltage through a resistor divider in non-isolated topologies (**Figure 27**). In DC mode, VRS ADC is configured to track the input voltage in real time. It enables DC mode feed-forward, which responds faster on line transient and allows primary voltage detection and protection prior to turning on the converter.

To configure VRSEN DC mode, set VRSEN ADC to general-purpose ADC mode (**vsp1_vrs_sel** = 0) and configure V_{IN} telemetry to "Non-pulsed/primary V_{IN} sense" (**tlmX_vin_src_sel** = 6). The test result of VRSEN DC mode feed-forward is shown in chapter 2.8.5.

To configure BVRSEN DC mode, set BVRSEN ADC to general-purpose ADC mode (**vsp2_vrs_sel** = 0) and configure V_{IN} telemetry to “Non-pulsed/primary V_{IN} sense” (**tlmX_vin_src_sel** = 7).

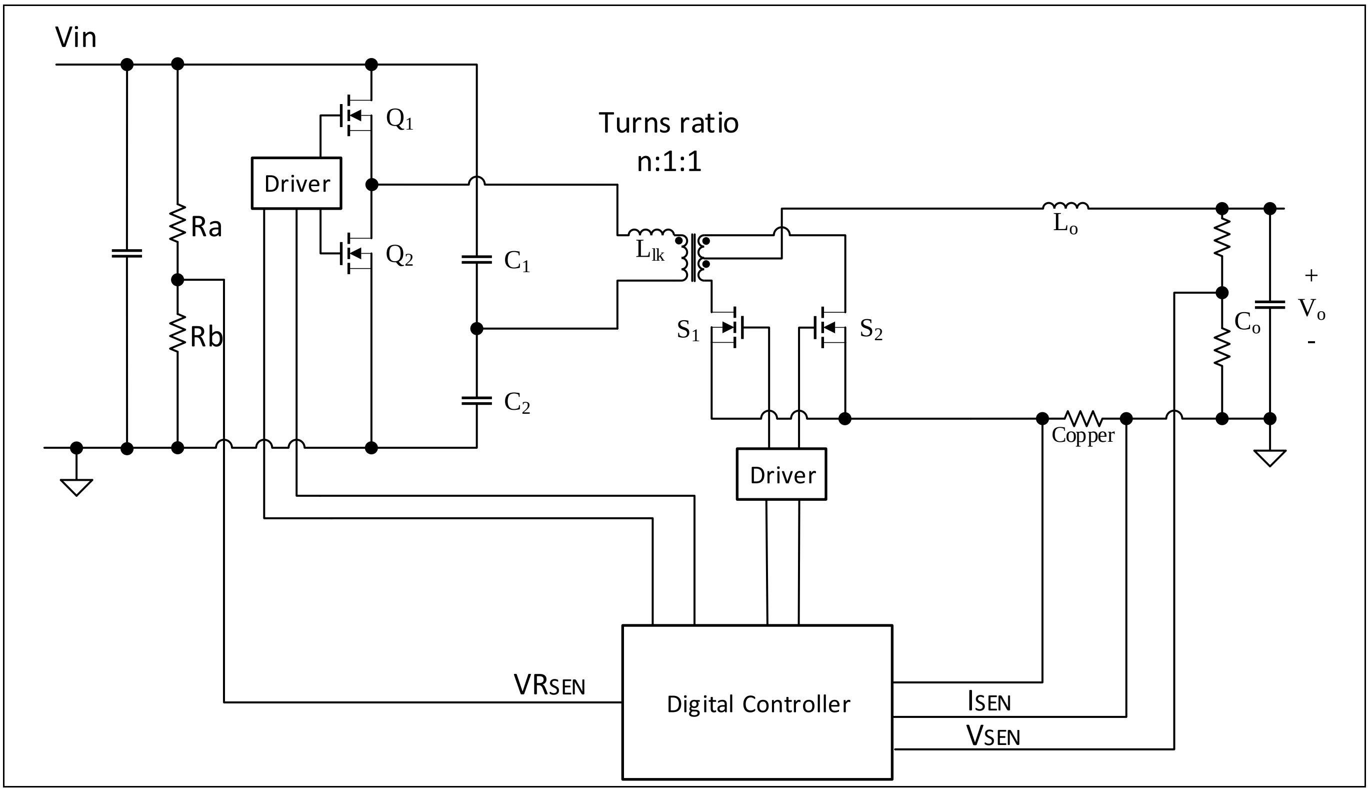


Figure 27 **VRSEN DC mode (non-isolated example)**

2.5 Feed-forward register descriptions

Table 7 describes the registers used by the feed-forward function.

Table 7 **Feed-forward relevant register descriptions**

Name	Address (loop 0/1)	Bits	Description
PID peripheral			
pid_ff_dt_adj	7000_1C00_H 7000_2000_H	[7:0]	Feed-forward offset adjustment for dead-time. Note, adjustment is always positive because dead-time always reduces duty-cycle. May also be used for general offset adjustment up to 25 percent duty-cycle. Computed as bridge topology: dt_adj = $T_{dead}/(T_{switch}/2)$ Non-bridge topology: dt_adj = T_{dead}/T_{switch} LSB = 2^-10, range = 0 to 0.2490 • Example: – Bridge topology, T_{dead} = 100 ns, T_{switch} = 4 µs – dt_adj = 100 ns/2000 ns = 0.05 – pid_ff_dt_adj = 0.05/2^-10 = 51
pid_kp_ff_lpf	7000_1C00_H	[12:8]	Feed-forward LPF coefficient. The computed feed-forward

Name	Address (loop 0/1)	Bits	Description
	7000_2000_H		term is passed through a LPF before being used by the PID as part of the duty-cycle calculation. • kp_exp = pid_ff_kp_lpf [4:2] • kp_man = 4 + pid_ff_kp_lpf [1:0] • kp = kp_man * 2^(kp_exp – 9) • F_{3db} (MHz) = [kp/(1 – kp)] * 50 MHz/2π Range = 62.7 kHz (0_D) to 23.87 MHz (26_D)
pid_ff_gain_scale	7000_1C00_H 7000_2000_H	[17:13]	Feed-forward gain scale. This parameter may be used to apply a gain to the computed feed-forward term. The typical setting is 16_D, which corresponds to a gain of 1.0. LSB = 2^-4, range = 0 to 1.9375
pid_ff_vrect_sel	7000_1C00_H 7000_2000_H	[19:18]	Feed-forward V_{RECT} source select. Feed-forward is computed as (V_{OUT}/V_{RECT}) for isolated and as (V_{OUT}/V_{IN}) for non-isolated buck-derived topologies. Note that the feed-forward computation may be over-ridden entirely using parameters pid_ff_override_sel and pid_ff_override (e.g., to override the HW computed feed-forward with a FW computation appropriate for boost or buck-boost derived topologies). 0 = VS1 ADC (VRSEN input) V_{RECT} 1 = VS2 ADC (BVRSEN input) V_{RECT} 2 = TS ADC (PRISEN input) V_{IN} 3 = pid_ff_vrect_override
pid_vrect_ref	7000_1C00_H 7000_2000_H	[27:20]	PID coefficient scaling reference voltage. PID coefficients are scaled with V_{RECT} to maintain a constant loop gain. This parameter defines the reference V_{RECT} voltage at which the gain scale is 1.0. This parameter should be set to the expected nominal V_{RECT} voltage and it should be set before optimization of PID coefficients K_p, K_i and K_d. Example: V_{in_nom} = 48 V, FB topology, N_{turn} = 3, pid0_vrect_ref = 48 V/3/0.32 V = 50 LSB = 0.32 V, range = 0.0 V to 81.6 V
pid_ff_vrect_override	7000_1C1C_H 7000_201C_H	[11:0]	V_{RECT} override for PID feed-forward computation in internal VS ADC format. Used only when selected by pid_ff_vrect_sel. LSB = 1.25 mV, range = 0 V to 5.11875 V **Compute from target V_{RECT}:** • pid_ff_vrect_override (U12.0) = (V_{RECT} (V)/1.25 mV) * MFR_VRECT_SCALE • Example: – V_{RECT} = 16 V – MFR_VRECT_SCALE = $B050_H$ = 0.078125 – pid_ff_vrect_override = 1000 **Compute from target V_{IN}:** • pid_ff_vrect_override (U12.0) = (V_{IN}(V)/1.25 mV) *

Name	Address (loop 0/1)	Bits	Description
			MFR_TRANSFORMER_SCALE * MFR_VRECT_SCALE • Example: – V_{IN} = 60 V – MFR_VRECT_SCALE = B050$_H$ = 0.078125 – MFR_TRANSFORMER_SCALE = B155$_H$ = 0.333 – pid_ff_vrect_override = 1249
pid_ff_override	7000_1C20$_H$ 7000_2020$_H$	[9:0]	PID feed-forward override value. Used when selected by pid_ff_override_sel. This parameter along with pid_ff_override_sel may be used to override the HW computed feed-forward with a FW computation appropriate for boost or buck-boost derived topologies. LSB = 2^-10, range = 0.999
pid_ff_override_sel	7000_1C20$_H$ 7000_2020$_H$	[10]	PID feed-forward override select. 0 = use computed feed-forward 1 = use pid_ff_override
Telem peripheral			
tlm_vin_src_sel	7000_3400$_H$ 7000_3800$_H$	[30:28]	Input voltage telemetry source select. 0 = VRSEN. Secondary V_{RECT} sense, **vrs_init** prior to start-up 1 = BVSEN_BVRSEN. Secondary V_{RECT} sense, **vrs_init** prior to start-up 2 = loop 0 V_{OUT}. Select on loop 1 when loop 1 V_{IN} provided by loop 0 V_{OUT} (e.g., post-buck) 3 = TS ADC V_{IN}. Non-pulsed/primary V_{IN} sense via telemetry ADC (PRISEN) 4 = tlm_vin_force. Forced V_{IN} via FW (e.g., FW override of HW computation by tlm_vin_force) 5 = VRSEN. Secondary V_{RECT} sense, **0 V** prior to start-up. Select on loop 1 when sharing loop 0 V_{RECT} sense 6 = VRSEN. Non-pulsed/primary V_{IN} sense 7 = BVSEN_BVRSEN. Non-pulsed/primary V_{IN} sense
Common peripheral			
vrs_cmp_wdt_thr	7000_3018$_H$	[9:0]	V_{RECT} comparator watchdog time-out threshold. The watchdog timer measures from the rising edge of the PWMs indicated in ceX_on_mask0 and ceX_on_mask1. If V_{RECT} has not tripped the comparator by the time the time-out threshold is reached, it is assumed V_{RECT} is below the comparator threshold and the V_{RECT} sense will enter its hold phase. This threshold should be set to a value greater than the expected time for V_{RECT} to go high but less than the tracking start threshold defined by vrs_track_start_thr. This threshold is shared by VRS1 and VRS2. LSB = 10 ns, range = 0 to 10230 ns

Name	Address (loop 0/1)	Bits	Description
vrs_track_start_thr	7000_3018_H	[19:10]	V_{RECT} tracking start time threshold. This threshold is compared against the same timer used by vrs_cmp_wdt_thr, which is started on the rising edge of the PWMs indicated in ceX_on_mask0 and ceX_on_mask1. When the timer exceeds this threshold the V_{RECT} sense moves from its hold phase to its tracking phase. This threshold should be set approximately 250 ns above the expected time of arrival of the V_{RECT} rising edge to insure adequate settling of the analog front end. This threshold is shared by VRS1 and VRS2. LSB = 10 ns, range = 0 to 10230 ns
vrect_div_2_sel	7000_3018_H	[20]	Defines equation used for rectification voltage (V_{RECT}) measurement. 0 = vrect_even + vrect_odd, for non-bridge topologies 1 = (vrect_even + vrect_odd)/2, for bridge topologies
vsp1_vrs_sel	7000_3018_H	[21]	VS1 (VRSEN) ADC VRS mode select. 0 = general-purpose ADC mode 1 = VRS mode
Vsp2_vrs_sel	7000_3018_H	[22]	VS2 (BVSEN_BVRSEN) ADC VRS mode select. 0 = V_{OUT} sense (VS) mode 1 = VRS mode
vrs_cmp_ref_sel	7000_3018_H	[27]	VRS comparator threshold select. This threshold is shared by VRS1 and VRS2. 0 = 500 mV 1 = 300 mV
vrs_voltage_init	7000_301C_H	[7:0]	Initial voltage for VS1 (VRSEN) and VS2 (BVSEN_BVRSEN) tracking integrators when operating in VRS mode. This setting must be greater than the rectification voltage corresponding to VIN_ON for controller start-up when a rectification voltage is selected for the input voltage telemetry input. LSB = 20 mV, range = 0 to 5.1 V **Compute from initial target V_{in_init}:** • vrs_voltage_init(U12.-4) = (V_{in_init}(V)/20 mV) * MFR_TRANSFORMER_SCALE*MFR_VRECT_SCALE • Example: – V_{in_init} = 48 V – MFR_TRANSFORMER_SCALE = 0.333469 – MFR_VRECT_SCALE = $B050_H$ = 0.078125 – vrs_voltage_init = 63
vrs_same_cycle_en	7000_301C_H	[8]	VRS same-cycle mode enable. When enabled, live V_{RECT} updates will begin vrs_meas_start_thr samples after entering tracking mode for faster feed-forward response. Otherwise V_{RECT} will only be updated on the falling PWM edge.

Name	Address (loop 0/1)	Bits	Description
			0 = same-cycle mode disabled 1 = same-cycle mode enabled
vrs_meas_start_thr	7000_301C$_H$	[16:12]	VRS same-cycle sample counts before going to live updates LSB = 4 samples, range = 0 to 124 samples
vrs_min_pw	7000_301C$_H$	[11:9]	VRS de-glitch pulse width. LSB = 20 ns, range = 0 to 140 ns

2.6 PMBus command descriptions

Table 8 describes the PMBus^{TM} commands relevant to the feed-forward function.

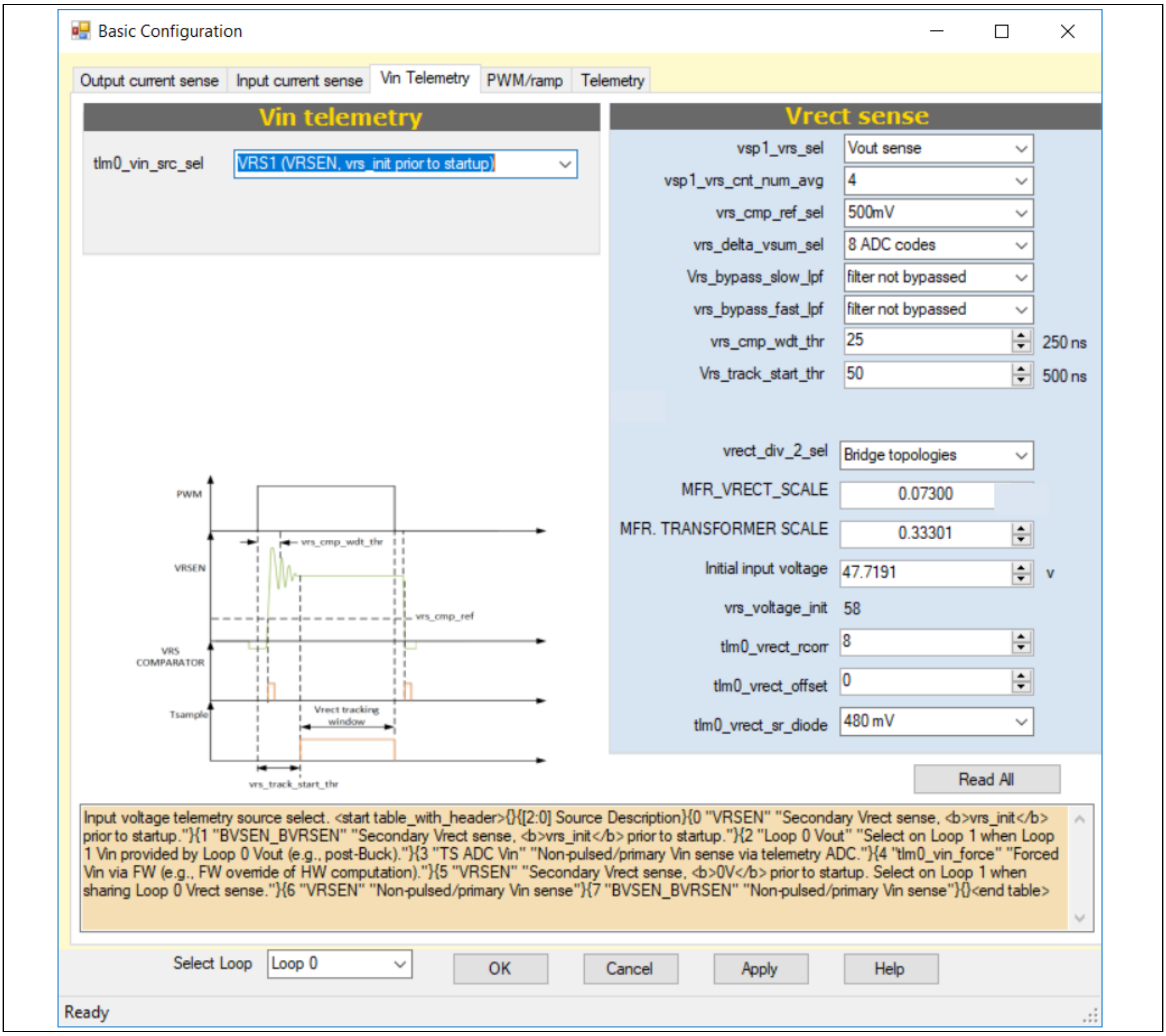
Table 8 Feed-forward relevant PMBus command descriptions

Command name	Command code	Format	Description
VOUT_SCALE_LOOP	29$_H$	LINEAR11	Scales VOUT_COMMAND and other V_{OUT} related commands for the external resistor divider at the voltage sense device input.
MFR_VRECT_SCALE	CD$_H$	LINEAR11	Defines the resistor divider scaling at the rectification voltage sense input. Should be set equal to (V_{rect_sense}/V_{RECT}).
MFR_TRANSFORMER_SCALE	CE$_H$	LINEAR11	Defines the transformer scaling ratio. Should be set equal to ($N_{turn\_secondary}/N_{turn\_primary}$).

2.7 XDPP1100 GUI design tool for V_{IN} sense and feed-forward

2.7.1 Input voltage sense and configuration

Figure 28 shows the XDPP1100 design tool for input voltage telemetry.



tlm0_vin_src_sel: VRS1 (VRSEN, vrs_init prior to startup)

vsp1_vrs_sel: Vout sense

vsp1_vrs_cnt_num_avg: 4

vrs_cmp_ref_sel: 500mV

vrs_delta_vsum_sel: 8 ADC codes

Vrs_bypass_slow_lpf: filter not bypassed

vrs_bypass_fast_lpf: filter not bypassed

vrs_cmp_wdt_thr: 25 250 ns

Vrs_track_start_thr: 50 500 ns

vrect_div_2_sel: Bridge topologies

MFR_VRECT_SCALE: 0.07300

MFR. TRANSFORMER SCALE: 0.33301

Initial input voltage: 47.7191 V

vrs_voltage_init: 58

tlm0_vrect_rcorr: 8

tlm0_vrect_offset: 0

tlm0_vrect_sr_diode: 480 mV

Input voltage telemetry source select. <start table_with_header>{}{[2:0] Source Description}{0 "VRSEN" "Secondary Vrect sense, <b>vrs_init</b> prior to startup."}{1 "BVSEN_BVRSEN" "Secondary Vrect sense, <b>vrs_init</b> prior to startup."}{2 "Loop 0 Vout" "Select on Loop 1 when Loop 1 Vin provided by Loop 0 Vout (e.g., post-Buck)."}{3 "TS ADC Vin" "Non-pulsed/primary Vin sense via telemetry ADC."}{4 "tlm0_vin_force" "Forced Vin via FW (e.g., FW override of HW computation)."}{5 "VRSEN" "Secondary Vrect sense, <b>0V</b> prior to startup. Select on Loop 1 when sharing Loop 0 Vrect sense."}{6 "VRSEN" "Non-pulsed/primary Vin sense"}{7 "BVSEN_BVRSEN" "Non-pulsed/primary Vin sense"}{}<end table>

Select Loop: Loop 0

Figure 28 GUI design tools – input voltage sensing

Use register **tlmX_vin_src_sel** to select the input voltage source. Here "X" is the loop number. X = 0 for loop 0, X = 1 for loop 1. Loop selection is located at the bottom of the design tool (**Figure 28**). Table 3 lists the options for input voltage source. The tool has a drop-down list for the user to select from.

2.7.1.1 VRS configuration

When the **tlm_vin_src_sel** register selects VRS1 or VRS2, the GUI enables the VRS configuration window to configure the following registers. The register name that starts with vsp1 is for VRS1 (VRSEN) and the register name that starts with vsp2 is for VRS2 (BVSEN_BVRSEN) configuration.

Table 9 VRS register configuration

Register name	Description
vspX_vrs_sel	VS(1 or 2), (VRSEN or BVRSEN) ADC VRS mode select. Configure the ADC to DC mode or VRS mode. For vsp1_vrs_sel, the non-pulsed DC mode is named as "general purpose ADC". For vsp2_vrs_sel, the DC mode is named as "vout sense". The VRS

Register name	Description
	mode is called “Vrect sense”.
vspX_vrs_cnt_num_avg	Defines the number of samples in the VRSEN dead-time measurement block averagers.
vrs_cmp_ref_sel	VRS comparator threshold select. It is not required for non-pulsed DC mode and the value will stay default for DC mode.
vrs_delta_vsum_sel	In the VRS tracking phase, the ADC output is filtered by both a fast (higher BW) LPF and a slow (lower BW) LPF. When the difference between the two LPF outputs is greater than the threshold selected by vrs_delta_vsum_sel, the fast filter output is selected, otherwise the slow filter output is selected.
vrs_bypass_slow_lpf	VRS slow filter bypass control.
vrs_bypass_fast_lpf	VRS fast filter bypass control.
vrs_cmp_wdt_thr	V_{RECT} comparator watchdog time-out threshold.
vrs_track_start_thr	V_{RECT} tracking start time threshold.
vrect_div_2_sel	Defines the equation used for rectification voltage (V_{RECT}) measurement.
vrs_voltage_init	Initial voltage for VS1 (VRSEN) and VS2 (BVSEN_BVRSEN) tracking integrators when operating in VRS mode.
vrs_min_pw	VRS de-glitch pulse width.

The vrs_voltage_init is calculated based on the transformer scale (MFR_TRANSFORMER_SCALE) and V_{RECT} resistor divider scale (MFR_VRECT_SCALE), so these two PMBus commands are also listed in the configuration tool (**Figure 28**).

When VRSEN or BVRSEN is used for V_{IN} telemetry, XDPP1100 offers additional configuration to compensate for the voltage drop introduced by parasitic resistance on the V_{RECT} sensing loop, and the voltage drop on the SR body diode when SR is in off mode. These registers are listed in Table 10.

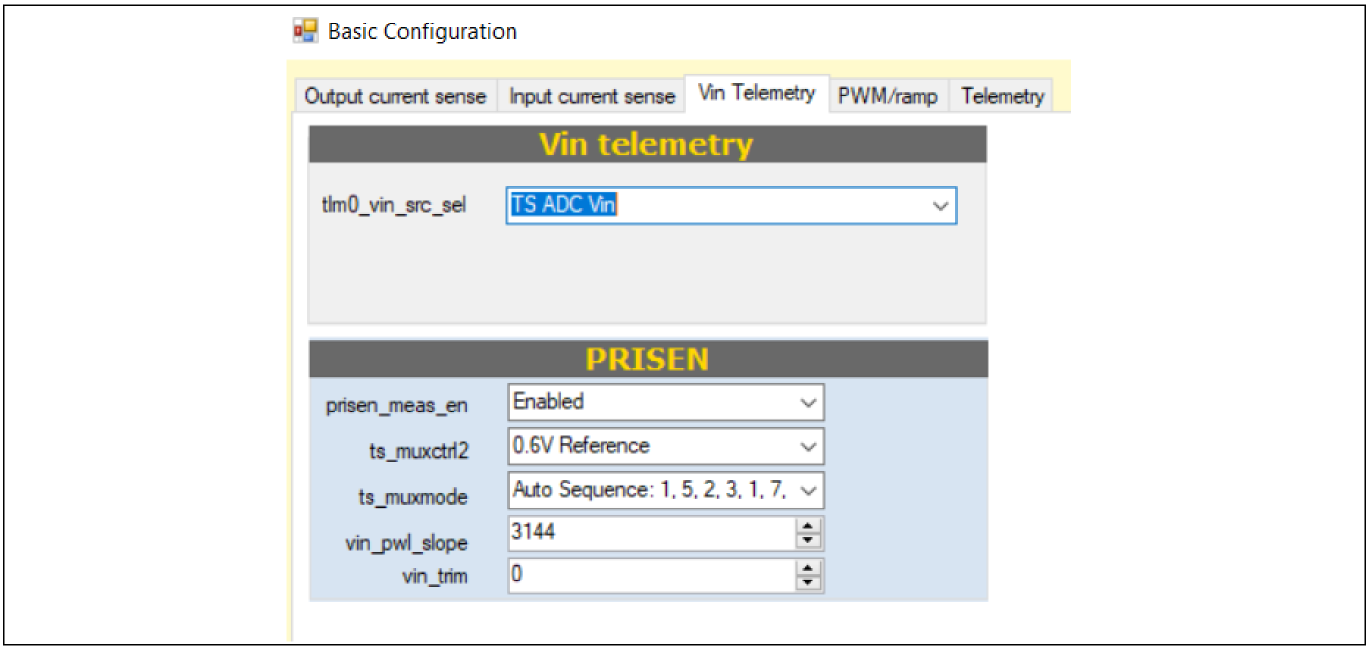
Table 10 VRS telemetry configuration

Register name	Description
tlmX_vrect_rcorr	Resistive correction term applied to rectification voltage (V_{RECT}) computation. This term is multiplied by the output current and added to the V_{RECT} value from the ADC. The user should enter the effective resistance as seen at the VRSEN input after the V_{RECT} sense resistor divider. LSB = 3.9 mΩ, range = 0 to 246 mΩ • Example: – At 48 V input, 40 V load, V_{IN} telemetry reads 47.5 V without compensation – The parasitic resistance is calculated by (48 V to 47.5 V)/40 V = 12.5 mΩ – tlm0_vrect_rcorr = 12.5/3.9 = 3
tlmX_vrect_offset	Voltage offset correction term applied to the rectification voltage (V_{RECT}) computation. LSB = 1.25 mV, range = -80 to 78.75 mV
tlmX_vrect_sr_diode	Voltage correction term on V_{RECT} computation when SR FETs are off. Intended to compensate for one or two series body diodes. 0 = 480 mV

Register name	Description
	1 = 640 mV 2 = 800 mV 3 = 960 mV 4 = 0 mV 5 = 1280 mV 6 = 1600 mV 7 = 1920 mV

2.7.1.2 TS ADC PRISEN configuration

When tlm_vin_src_sel register is set to "TS ADC", the GUI enables the PRISEN configuration window to configure the registers in Table 11.



tlm0_vin_src_sel: TS ADC Vin

prisen_meas_en: Enabled

ts_muxctrl2: 0.6V Reference

ts_muxmode: Auto Sequence: 1, 5, 2, 3, 1, 7,

vin_pwl_slope: 3144

vin_trim: 0

Figure 29 GUI design tool – V_{IN} source is TS ADC

Table 11 PRISEN register configuration

Register name	Description
prisen_meas_en	TS ADC PRISEN measurement enable. It should be set to 1 to enable PRISEN. When enabled, the TS ADC will measure the PRISEN input when selected by ts_muxmode and tx_muxctrl2.
ts_muxctrl1	TS ADC MUX1 input source select. The output of MUX1 is connected to MUX2 input 7. The most common setting of this register is 0 to measure the internal temperature of the controller.
ts_muxctrl2	TS ADC MUX2 input source select. The output of MUX2 is connected to the TS ADC input. 0 = 0.6 V reference (test only) 1 = PRISEN

Register name	Description
	2 = IMON 3 = ATSEN 4 = BTSEN 5 = XADDR1 unfiltered 6 = XADDR2 unfiltered 7 = MUX1 output
ts_muxmode	TS ADC input sequence control. When bit [2] is 0, the TS ADC input is entirely determined by the settings of ts_muxctrl1 and ts_muxctrl2. When bit [2] is 1, MUX2 auto-sequences its input using the pattern in the table below. If the sequence includes MUX2 input 7 (MUX1), the source in this time slot is determined by the setting of ts_muxctrl1. 0 to 3 = defined by ts_muxctrl1, 2 4 = auto sequence: 1, 2, 1, 3, 1, 2, 1, 4 5 = auto sequence: 1, 5, 1, 3, 1, 7, 1, 4 6 = auto sequence: 1, 5, 2, 3, 1, 7, 2, 4 7 = auto sequence: 1, 5, 2, 3, 1, 7, 6, 4
vin_pwl_slope	TS ADC PRISEN input voltage (V_{IN}) piecewise linear slope term. LSB = 2^-14 V/V, range = 0 to 0.24994 V/V
vin_trim	TS ADC PRISEN input voltage (V_{IN}) offset term

The **vin_pwl_slope** assumes a linear slope of the PRISEN signal, which is proportional to input voltage V_{IN}. TS ADC resolution is 2.344 mV (1.2 V/2^9). The **vin_pwl_slope** can be calculated by the following equation:

$$vin\_pwl\_slope = \frac{\Delta Vin \times 1.2 \times 2^5}{\Delta VPRISEN} \tag{2.8}$$

To utilize the full input voltage range of TS ADC, it is recommended to set the scale of V_{IN} resistor divider per:

$$\frac{1.2V}{Vin\_max}$$

For example, a 48 V DC-DC brick with maximum 96 V input voltage. The V_{IN} resistor divider ratio is set to 1.2 V/96 V = 0.0125 V/V. There is no offset added to the resistor divider (**vin_trim** = 0).

$$vin\_pwl\_slope = \frac{96V \times 1.2 \times 2^5}{1.2V} = \frac{1 \times 1.2 \times 2^5}{0.0125} = 3072$$

2.7.1.3 FW forced input voltage

When the **tlm_vin_src_sel** register is set to 4, the GUI enables **tlm_vin_force** configuration. The vin_force is typically used during debugging to emulate input voltage to prevent power supply shutdown due to VIN_UV or VIN_OFF. Or it is used by the FW to override the measured input voltage. For this reason, this register will not be stored in OTP memory when the user stores the configuration. This information can be found in the register map document. If the column OTP value is “false”, it means the register will not be stored in OTP memory. The OTP “false” register will also not be stored in RAM by the “write all” shortcut button when loading a design file through the GUI. Please go to the register map and manually write the register to make the value copy to RAM.

Table 12 V_{IN} force register

Register name	Description
tlmX_vin_force	Forced V_{IN} input value. LSB = 62.5 mV, range = 0.0 to 127.9375 V • Example: – Force loop 0 input voltage to 48 V – tlm0_vin_force = 48 V/62.5 mV = 768

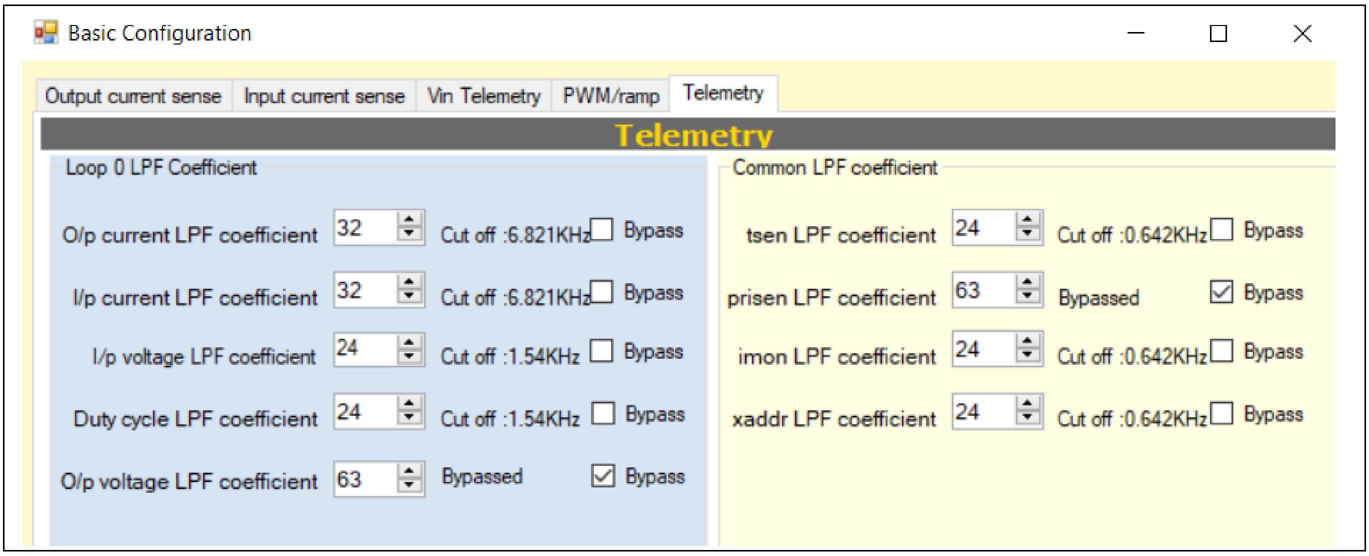
2.7.1.4 V_{IN} telemetry filter

The input voltage telemetry LPF is defined by **tlmX_kfp_vin**.

Table 13 V_{IN} telemetry LPF

Register name	Description
tlmX_kfp_vin [5:0]	Input voltage telemetry LPF coefficient index. Clamped to 39_D. • kfp_exp = tlm_kfp_vin [5:2] • kfp_man = 4 + tlm_kfp_vin [1:0] • kfp = kfp_man * 2^kfp_exp * 2^-13 • F3db (kHz) = [kfp/(1 - kfp)] * F_{switch} (kHz)/2π Range from 0.019 kHz to 30.947 kHz. • Example: – tlmX_kfp_vin = 24_D = 011000_B – kfp_exp = 6, kfp_man = 4+ 0 = 4 – kfp = 4 * 2^6 * 2^-13 = 0.03125 – F_{switch} = 250 kHz – F3db = (0.03125/(1-0.03125))*250 kHz/2π = 1.2835 kHz

All the telemetry LPFs can be configured in the design tool “Basic Configuration” and “Telemetry” tab. In **Figure 30**, the loop 0 “I/p voltage LPF coefficient” defines the **tlm0_kfp_vin** register.



Loop 0 LPF Coefficient

O/p current LPF coefficient 32 Cut off :6.821KHz □ Bypass

I/p current LPF coefficient 32 Cut off :6.821KHz □ Bypass

I/p voltage LPF coefficient 24 Cut off :1.54KHz □ Bypass

Duty cycle LPF coefficient 24 Cut off :1.54KHz □ Bypass

O/p voltage LPF coefficient 63 Bypassed ☑ Bypass

Common LPF coefficient

tsen LPF coefficient 24 Cut off :0.642KHz □ Bypass

prisen LPF coefficient 63 Bypassed ☑ Bypass

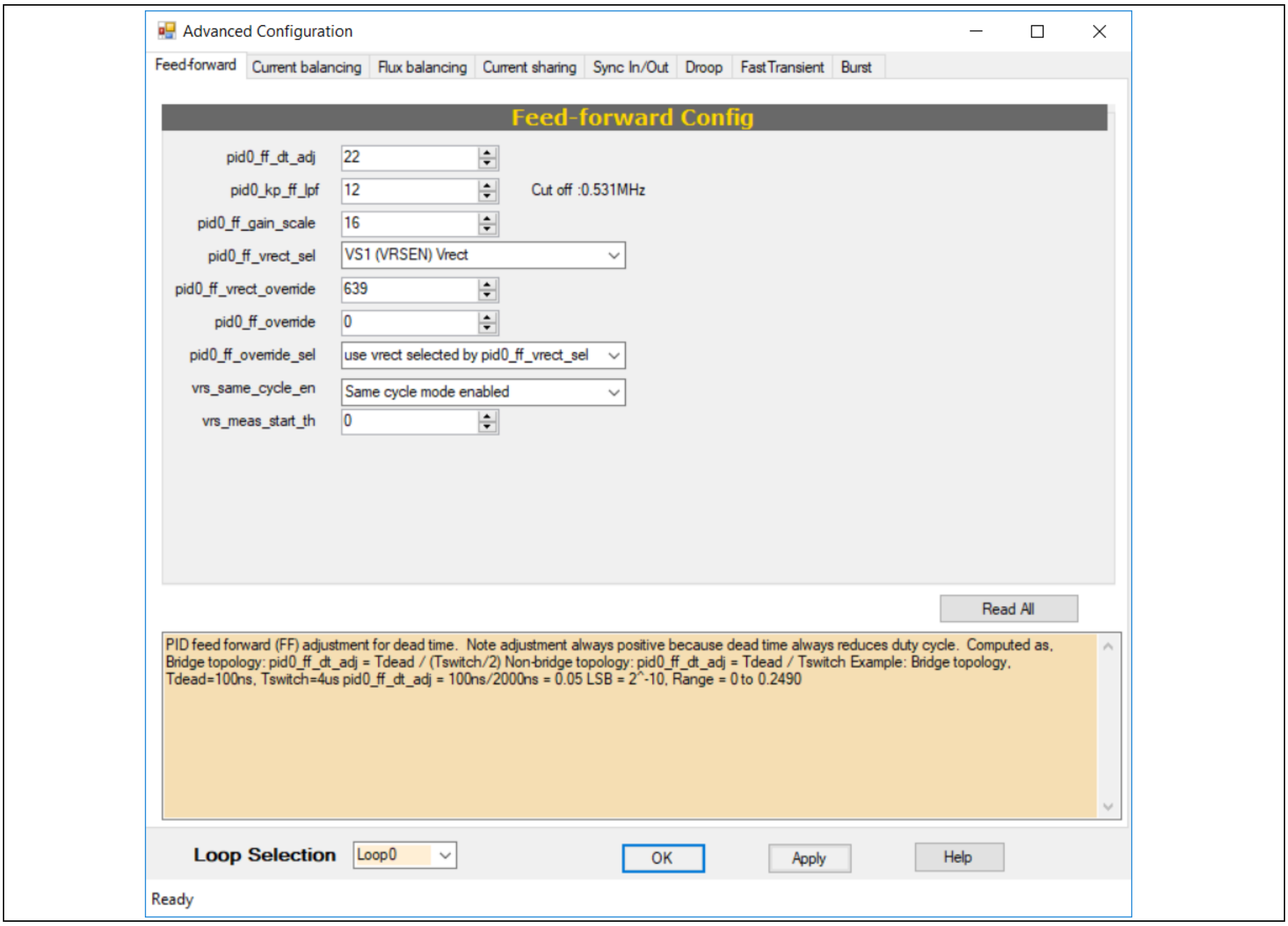
imon LPF coefficient 24 Cut off :0.642KHz □ Bypass

xaddr LPF coefficient 24 Cut off :0.642KHz □ Bypass

Figure 30 GUI design tool – telemetry LPF

2.7.2 Feed-forward configuration

A feed-forward configuration tool is included in the Advanced Configuration tool in the GUI. The tool guides the user to configure the registers shown in **Figure 31**. Detailed descriptions of each register and examples can be found in chapter 2.4 and chapter 2.5.



pid0_ff_dt_adj 22
pid0_kp_ff_lpf 12 Cut off :0.531MHz
pid0_ff_gain_scale 16
pid0_ff_vrect_sel VS1 (VRSEN) Vrect
pid0_ff_vrect_override 639
pid0_ff_override 0
pid0_ff_override_sel use vrect selected by pid0_ff_vrect_sel
vrs_same_cycle_en Same cycle mode enabled
vrs_meas_start_th 0

PID feed forward (FF) adjustment for dead time. Note adjustment always positive because dead time always reduces duty cycle. Computed as, Bridge topology: pid0_ff_dt_adj = Tdead / (Tswitch/2) Non-bridge topology: pid0_ff_dt_adj = Tdead / Tswitch Example: Bridge topology, Tdead=100ns, Tswitch=4us pid0_ff_dt_adj = 100ns/2000ns = 0.05 LSB = 2^-10, Range = 0 to 0.2490

Loop Selection Loop0

Figure 31 GUI design tools – input voltage feed-forward

2.8 Test result of V_{RECT} sensing-based feed-forward

The start-up and line transient performance are tested with the REF_600W_FBFB_XDPP1100 reference design, which is an isolated quarter-brick converter that converts 48 V to 12 V with FB-FB topology. It is configured to operate in VMC. The document of the reference design is available online at **REF_600W_FBFB_XDPP1100**.

Figure 32 is the schematic of the power stage of the 12 V/50 A FB-FB converter. Primary MOSFETs are OptiMOS^{TM} 100 V/5 mΩ BSC050N10NS5s. Infineon isolated gate driver **2EDF7275K** is used to drive the primary MOSFETs. The **2EDF7275K** provides 1.5 kV functional isolation and 4 A/8 A gate driver capability. The secondary synchronous rectifier (SR) uses two 40 V/1 mΩ BSC010N04LS6s in parallel at each location. The SR MOSFETs are also driven by **2EDF7275K**s to simplify the BOM.

A planar transformer is used for the lowest board profile. The transformer turns ratio is 3:1.

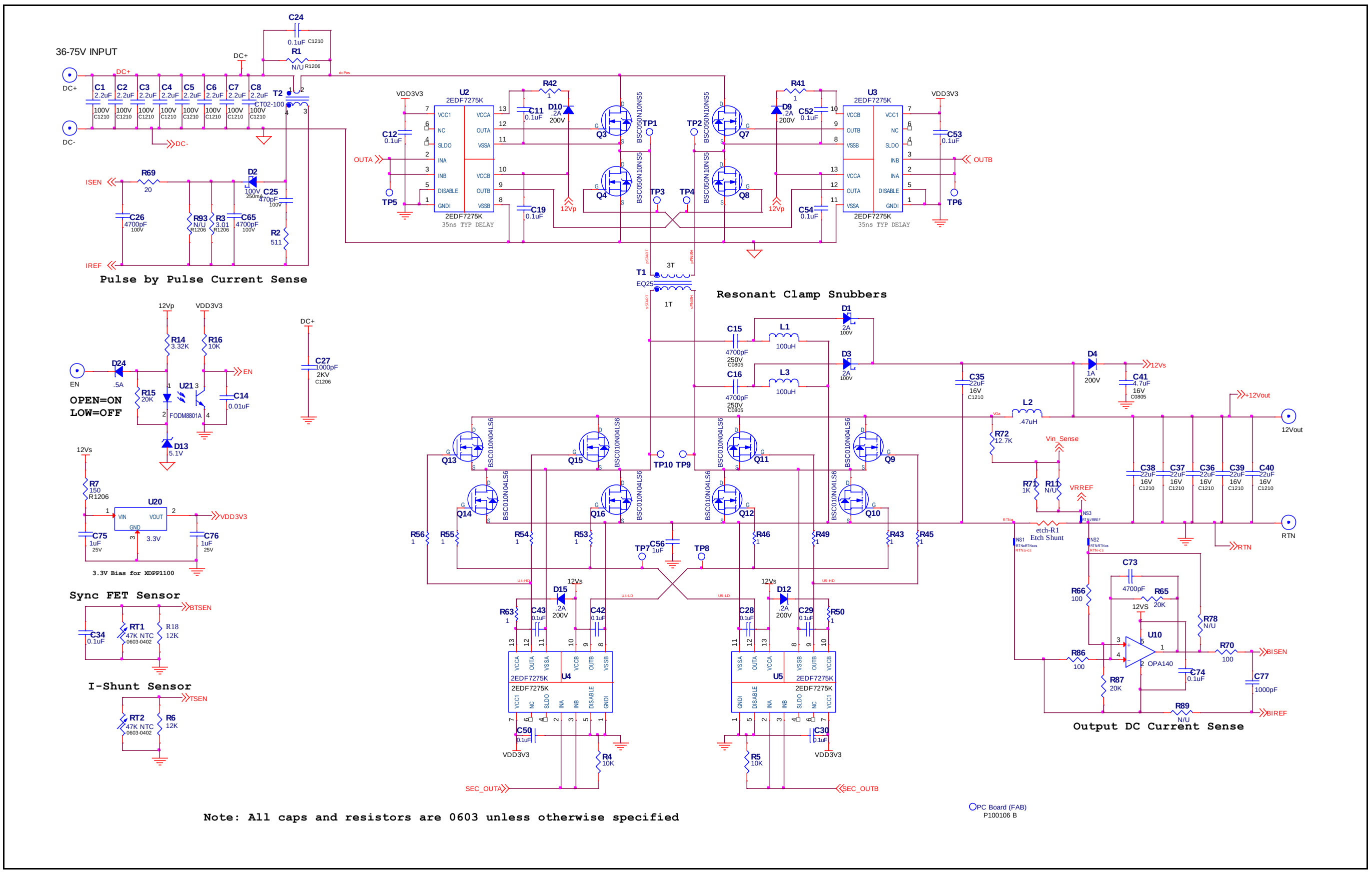


Figure 32 Schematic of 600 W FB-FB power stage

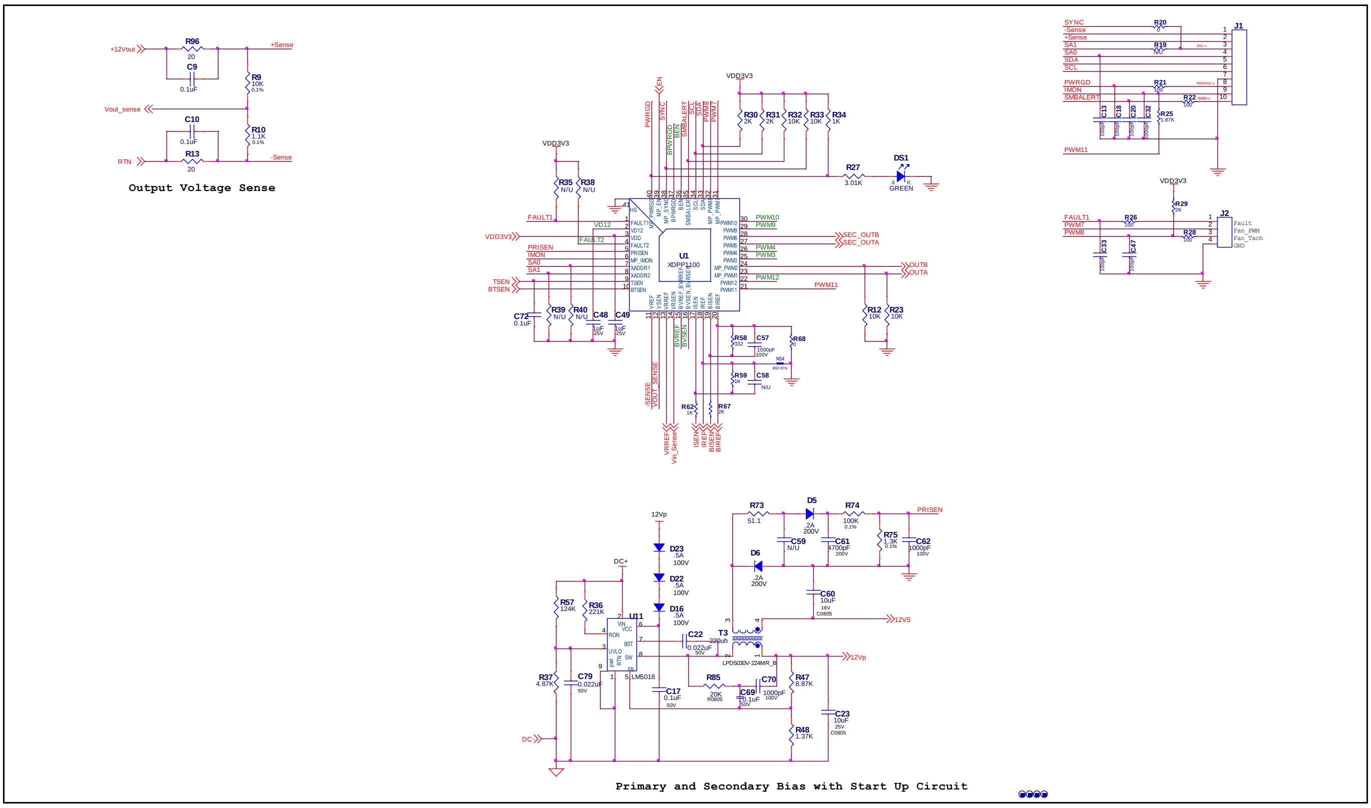


Figure 33 Schematic of the FB-FB control circuit

The schematic of the control circuit and auxiliary power supply is shown in **Figure 33**. The VRSEN/VRREF is connected to sense the input voltage through transformer secondary winding. The V_{RECT} sensing enables fast feed-forward response and flux balancing. There is another input voltage sensing circuit at the auxiliary transformer T3. It is connected to the PRISEN pin and provides input voltage telemetry while the main converter is not in operation.

2.8.1 FB-FB feed-forward configuration

Table 14 FB-FB VMC feed-forward register configuration

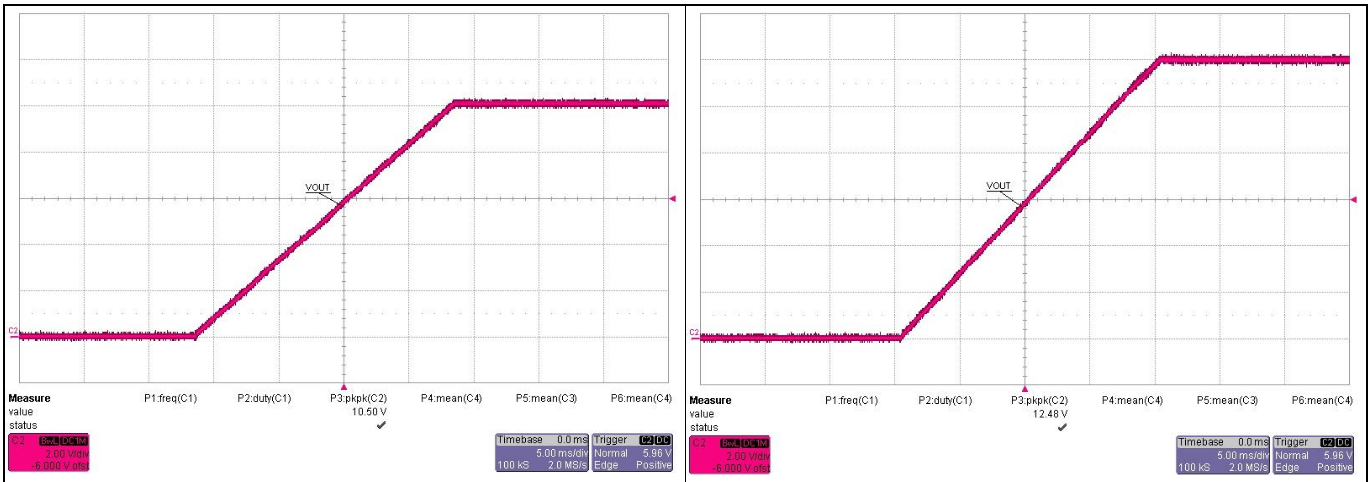
Register name	Register value	Meaning
pid_ff_dt_adj	22	Feed-forward offset adjustment for dead-time. 22*2^-10 = 2.15 percent
pid_kp_ff_lpf	12	Feed-forward LPF = 0.53 MHz
pid_ff_gain_scale	16	Feed-forward gain = 1
pid_ff_vrect_sel	0	Feed-forward V_{RECT} source select = VS1 ADC (VRSEN input)
pid_vrect_ref	50	Nominal input is 48 V, transformer turns ratio is 3:1. 48/3/0.32 = 50
pid_ff_override_sel	0	Feed-forward source select. 0 = use computed feed-forward
tlm_vin_src_sel	0 and 3	Input voltage telemetry source select. 0 = VRSEN. Secondary V_{RECT} sense, **vrs_init** prior to start-up 3 = PRISEN. Non-pulsed V_{IN} sense via telemetry ADC FW patch determines the tlm_vin_src based on the on-/off-state
vsp1_vrs_sel	1	VS1 (VRSEN) ADC VRS mode. 1 = VRS mode
vrs_cmp_ref_sel	0	VRS edge comparator threshold is 500 mV
vrs_cmp_wdt_thr	25	VRS comparator watchdog timer set to 250 ns
vrs_track_start_thr	50	VRS tracking start time set to 500 ns
vrect_div_2_sel	1	Bridge topology
vrs_voltage_init	58	Initial voltage for VS1 (VRSEN input) tracking = 48 V
vrs_same_cycle_en	1	Same-cycle mode enabled
vrs_min_pw	0	VRS edge de-glitch PW is 0 ns
vrs_meas_start_thr	0	The number of samples after vrs_track_start_thr is set to 0 ns

Table 15 FB-FB VMC feed-forward PMBus command value

PMBus name	PMBus value	Description
VOUT_SCALE_LOOP	0.0992126465	Calculated by R10/(R9+R10)
MFR_VRECT_SCALE	0.072754	Calculated by R71/(R71+R72)
MFR_TRANSFORMER_SCALE	0.333008	Calculated by N_S/N_P

2.8.2 Start-up behavior

Start-ups are tested at no load and full load over input voltage range. **Figure 34** shows waveforms at 50 A full load at the minimum and the maximum input voltage. The output voltage is set to 10 V at 36 V input for close loop regulation. The V_{OUT} ramp is clean, without glitches.



36 V input, 50 A load
At 36 V input, output voltage is set to 10 V

72 V input, 50 A load
Output voltage is 12 V

Figure 34 Monotonic start-up (2 V/div)

2.8.3 Pre-bias start-up

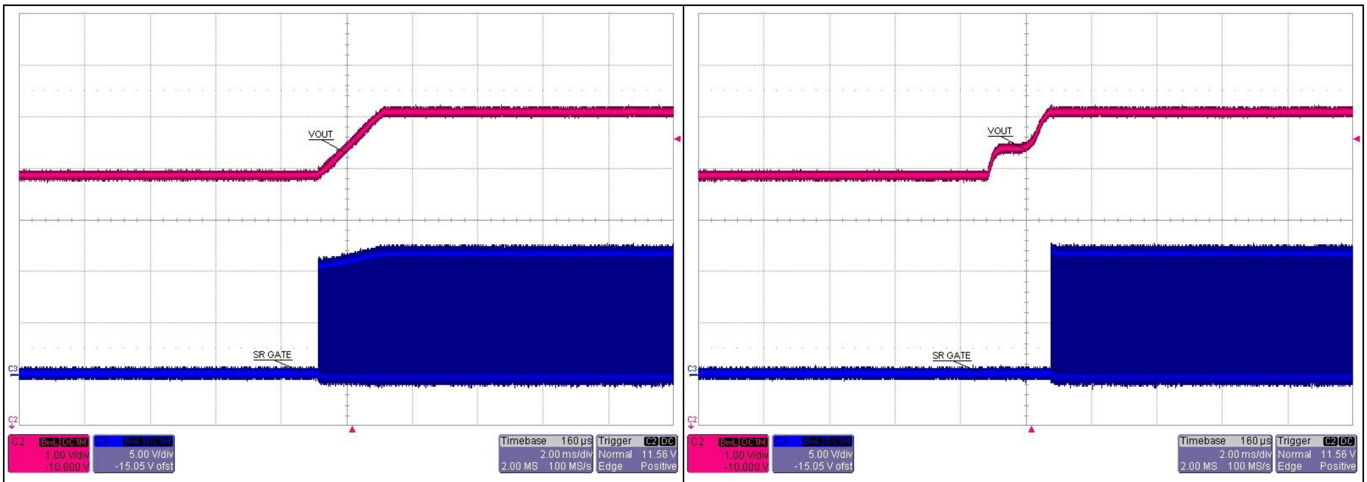
90 percent (10.8 V) pre-bias start-up is tested with SR mode and DE mode. The input voltage is measured by the TS ADC through PRISEN when the FB converter is not in regulation. The pre-bias start-up is achieved by measuring V_{IN} and V_{OUT} prior to start-up. The initial duty-cycle is calculated by the feed-forward circuit based on the V_{IN} and V_{OUT}, then the PID control loop takes over for the rest of the start-up ramp.

The start-up ramp retains the same slope as regular start-up from 0 V. The rise time is calculated by FW:

Ramp_time = TON_RISE * abs(VOUT_COMMAND - Prebias_voltage)/VOUT_COMMAND

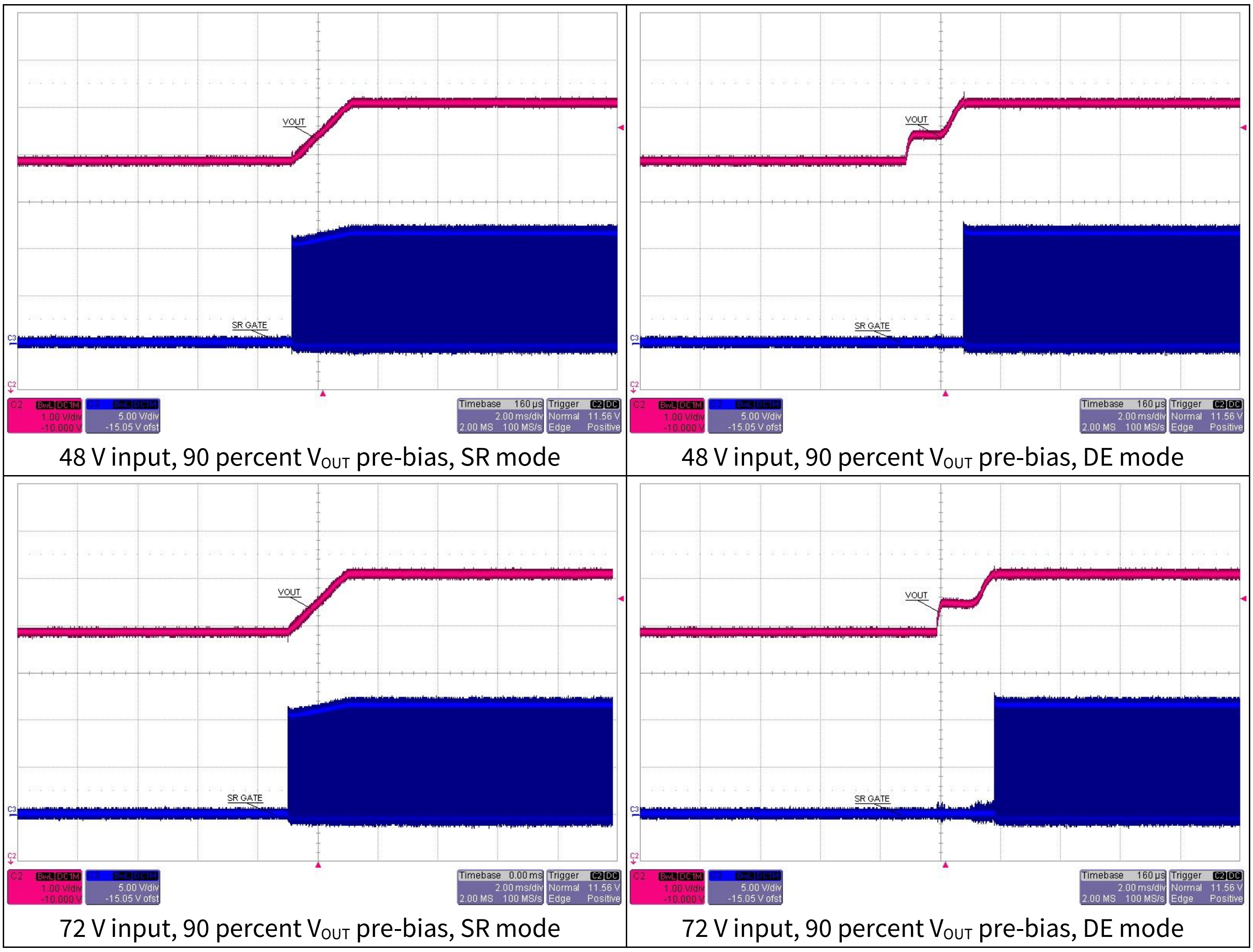
For example, TON_RISE = 20 ms, pre-bias is set to 90 percent of VOUT_COMMAND, the rise time is 2 ms.

Ch2: V_{OUT} (1 V/div, offset = -10 V), Ch4: SR gate drive (5 V/div)



42 V input, 90 percent V_{OUT} pre-bias, SR mode

42 V input, 90 percent V_{OUT} pre-bias, DE mode



48 V input, 90 percent V_{OUT} pre-bias, SR mode

48 V input, 90 percent V_{OUT} pre-bias, DE mode

72 V input, 90 percent V_{OUT} pre-bias, SR mode

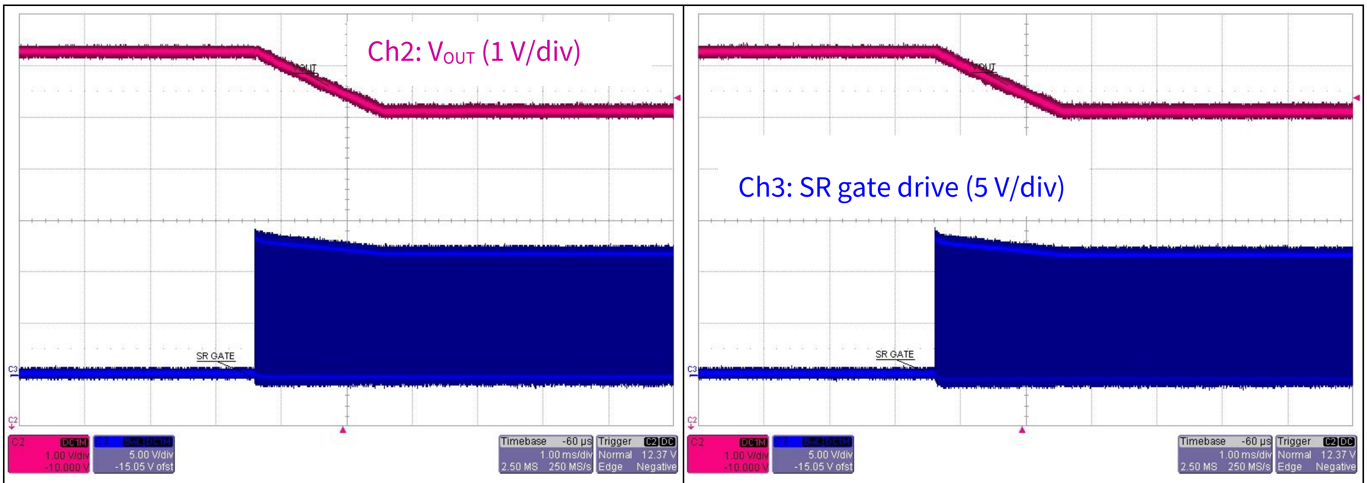
72 V input, 90 percent V_{OUT} pre-bias, DE mode

Figure 35 Pre-bias start-up waveforms with pre-bias voltage 10.8 V and no load

Figure 35 shows 10.8 V pre-bias start-up at 42 V, 48 V and 72 V input voltage. In SR mode, the V_{OUT} ramp is smooth (without glitches) over the input voltage range, indicating the feed-forward duty-cycle is accurate and meets system conditions.

The DE mode start-up shows a voltage step at the beginning. This happens in DE mode at no load, where the desired duty-cycle is less than feed-forward duty-cycle due to no negative current circulating in the SR MOSFET. The PID compensator would reduce duty-cycle until the voltage error approaches zero and resumes normal start-up. In DE mode start-up, the SR gate drive (blue trace in **Figure 35**) is enabled when V_{OUT} reaches the target window, which is defined by VOUT_COMMAND – Vout_target_window. The Vout_target_window is configured by PMBus command MFR_CONFIG_REGULATION bit [31:24]. The V_{OUT} waveform does not show obvious glitches when the SR is enabled.

The 110 percent (13.2 V) pre-bias start-up is tested at 42 V and 72 V input. When the pre-bias voltage is higher than target V_{OUT}, the SR is always enabled to discharge the pre-bias voltage, ignoring the DE start-up configuration. The PRISEN V_{IN} measurement will be slightly affected by the 13.2 V pre-bias voltage due to D4. This doesn't affect the performance of the pre-bias start-up.



42 V input, 110 percent V_{OUT} pre-bias

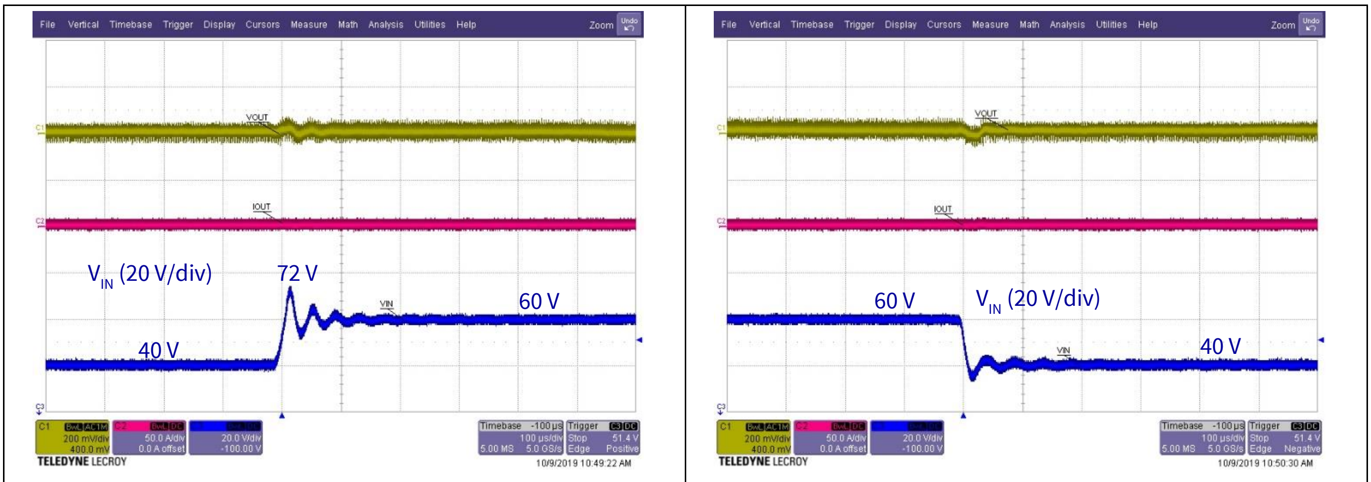
72 V input, 110 percent V_{OUT} pre-bias

Figure 36 **Pre-bias start-up waveforms with pre-bias voltage 13.2 V**

2.8.4 V_{IN} transient and feed-forward

Line transient is tested at no load and full load. Set **pid0_ff_vrect_sel** = 0 to select V_{RECT} as the source of input feed-forward sensing. The same-cycle response is enabled. Overshoot and undershoot are less than 100 mV in the line transient test. A Kikusui PCR4000LE power source is used for fast dv/dt slew rate. The input LC filter (100 µF + 1 µH + 100 µF) on the test fixture was removed to allow fast dv/dt. There is some ringing during V_{IN} transient due to the resonance between cable inductance and power supply input capacitor.

Ch1 = V_{OUT} AC coupled (200 mV/div), Ch2 = I_{OUT} (50 A/div), Ch3 = V_{IN} (20 V/div)

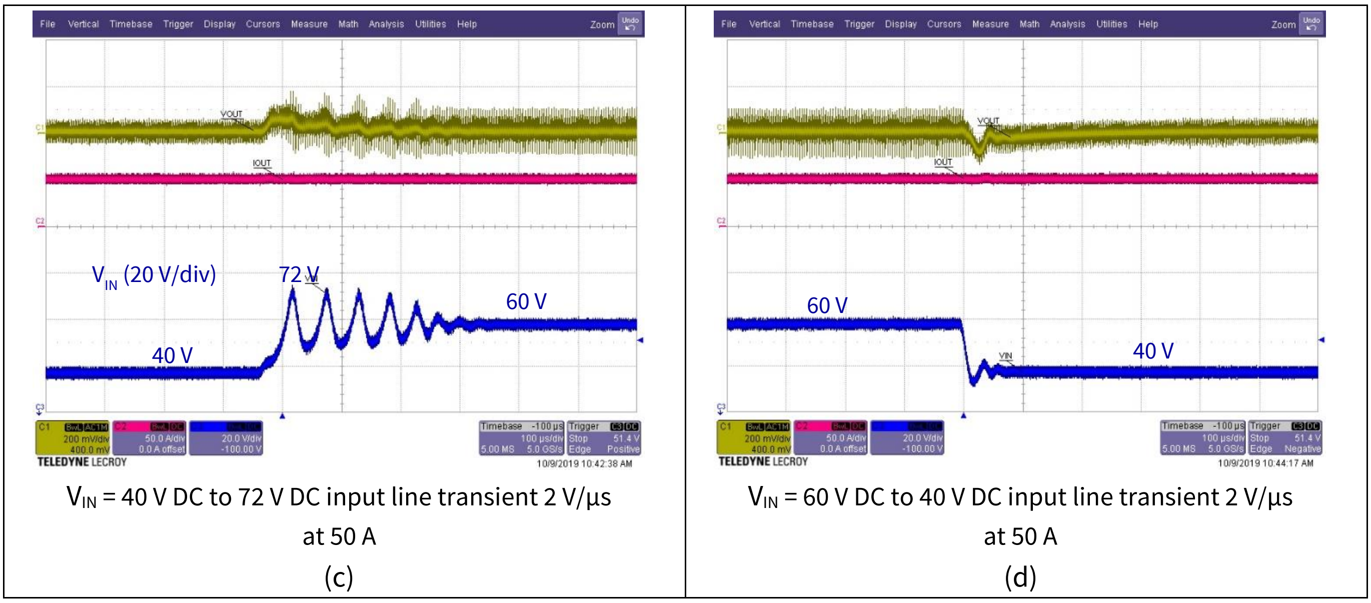


V_{IN} = 40 V DC to 72 V DC input line transient 2 V/µs at 0 A

(a)

V_{IN} = 60 V DC to 40 V DC input line transient 2 V/µs at 0 A

(b)



V_{IN} = 40 V DC to 72 V DC input line transient 2 V/µs at 50 A

(c)

V_{IN} = 60 V DC to 40 V DC input line transient 2 V/µs at 50 A

(d)

Figure 37 Line transient waveform (2 V/µs)

2.8.5 Same-cycle feed-forward

Comparison of the same-cycle feed-forward and standard feed-forward is tested with an ACF power supply with 3.3 V/30 A output. The results are shown in **Figure 38**. Enabling the same-cycle response reduces overshoot and undershoot by more than half during fast line transient. Set the **vrs_same_cycle_en** register to 1 to enable VRS same-cycle mode. Live V_{RECT} updates will begin after entering tracking mode and feed-forward duty-cycle is calculated in real time.

Register settings to enable same-cycle feed-forward: **vrs_same_cycle_en** = 1, **vrs_meas_start_th** = 0

Register settings to disable same-cycle feed-forward: **vrs_same_cycle_en** = 0, **vrs_meas_start_th** = 0

Ch1: V_{IN} (10 V/div), Ch3: PWM output for primary gate (2 V/div)

Ch4: V_{OUT} (50 mV/div) – the same-cycle feed-forward is enabled.

M4: V_{OUT} (50 mV/div) – the same-cycle feed-forward is disabled.

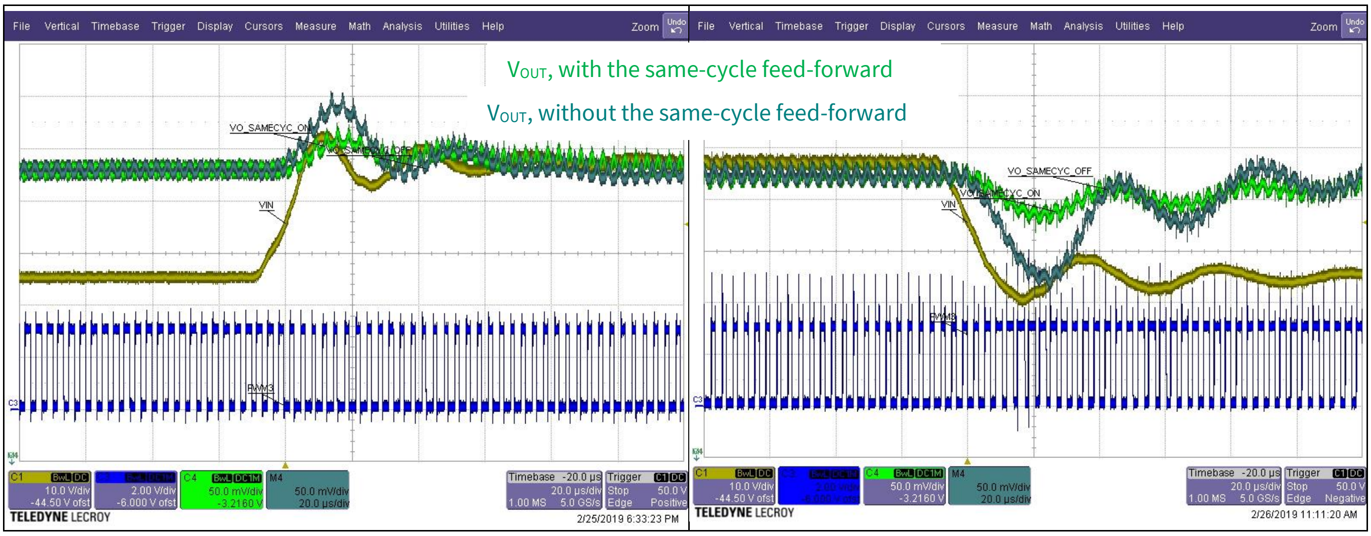


Figure 38 40 V to 60 V line transient under no-load condition

2.8.6 VRS sense DC mode

To verify the capability of VRS DC mode, a resistor divider is added to the input voltage and connected to the VRSEN pin of the FB-FB demo board. The original VRSEN resistors are disconnected. Isolation is ignored in this test and the primary ground and secondary ground are shorted together. Set the **vsp1_vrs_sel** = 0 to configure VRSEN ADC to the general ADC mode. Set **tlm0_vin_src_sel** = 6 for non-pulsed primary V_{IN} sense.

When VRSEN is connected to the primary resistor divider (**Figure 27**), MFR_VRECT_SCALE should be calculated by:

$$MFR\_VRECT\_SCALE = \frac{Rb}{(Ra+Rb)} \cdot \frac{k}{MFR\_TRANSFORMER\_SCALE} \qquad (2.8)$$

MFR_TRANSFORMER_SCALE is transformer turns ratio, equals to N_S/N_P.

k = 2 for HB topology, otherwise *k* = 1.

The comparison of the system response is shown in **Figure 39**.

Ch1: V_{IN} (10 V/div), M1: V_{IN} (10 V/div), Ch2: V_{IN} sense after the resistor divider (500 mV/div).

Ch3: PWM output for primary gate (5 V/div).

Ch4: V_{OUT} AC coupled (100 mV/div), VRS DC mode.

M4: V_{OUT} AC coupled (100 mV/div), VRS standard mode (none DC mode, not same-cycle response).

Register settings to enable VRS DC mode: **vsp1_vrs_sel** = 0, **tlm0_vin_src_sel** = 6 (loop 0)

Register settings to disable VRS DC mode: **vsp1_vrs_sel** = 1, **tlm0_vin_src_sel** = 0 (loop 0)

The V_{OUT} overshoot in VRS DC mode is reduced to 100 mV and it is similar to the same-cycle feed-forward result shown in chapter 2.8.4. It is about half of the overshoot of standard feed-forward.

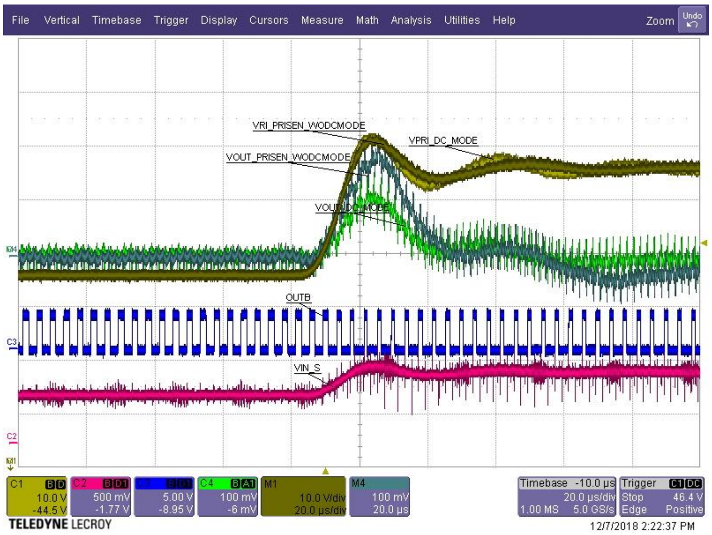


Figure 39 **FB-FB, 40 V to 70 V line transient (overshoot reduced from 180 mV to 100 mV)**

The test results of chapter 2.8.5 and chapter 2.8.6 tell us the feed-forward performance can be improved by enabling the same-cycle response. Within a certain input dv/dt range, the same-cycle feed-forward behaviors are as good as the VRS DC mode feed-forward. The VRS DC mode might show better response when the V_{IN} dv/dt is higher than 2 V/µs.

3 Current sense

In power supply design, curent sense (CS) plays an important role for control and protections. It is used for current telemetry, protection and current mode control. On top of that, the output current is often used for many other features such as current sharing, current balancing, burst mode control in light load condition, fan speed control, and DE control. CS accuracy is required to achieve the best performance of these features. Low latency is required for the current protection for system robustness. This chapter describes the XDPP1100 CS method and the advantages of the method.

3.1 CS requirements

Measuring current through a shunt resistor is the most common solution. For low-current applications, a fixed-resistance high-precision resistor can be used for CS. In high-current applications, a low-ohmic sense resistor should be used to limit the power loss on the shunt. This results in a smaller voltage signal on the sense resistor and could affect the accuracy of CS. Instead of increasing the value of the shunt resistor, amplifying the sense signal is a better solution.

PCB trace current sensing or DCR current sensing is another option besides the precision shunt resistor. DCR is the DC resistance of an inductor. There are several advantages of using PCB trace or DCR for current sensing, including no external components being required, there being no additional voltage drop and power loss on the shunt resistor, reduced layout footprint, and reduced system BOM cost. The disadvantage of the PCB copper and DCR current sensing is that the resistance has much larger variation than the precision shunt resistor, and the resistance varies with temperature. To use the PCB copper and DCR current sensing, the controller must able to trim the gain of the CS from board to board, and must have the capability for temperature compensation.

In an isolated power supply using a digital controller at the secondary side, the primary current is usually sensed via a current transformer. The current transformer typically reduces the current level by a 50:1 or 100:1 ratio, thus allowing the use of a low-power rated resistor at the secondary side of the transformer.

Table 16 lists the main methods of current sensing and their features.

Table 16 Current sensing methods

Sensing method	Advantages	Disadvantages	Applications
Shunt resistor	• Most common solution • Simple	• BOM cost • Additional power dissipation	• All applications
PCB trace	• No additional BOM cost • No extra power dissipation	• Higher temperature variation must be compensated • Need onboard trimming for accuracy • Potential for higher noise-to-signal ratio, layout is critical	• High current applications
PCB trace with amplifier	• No extra power dissipation	• Higher temperature variation must be compensated • Need onboard trimming for accuracy • External amplifier is required	• High current applications

Sensing method	Advantages	Disadvantages	Applications
Current transformer	• Simple method	• Additional BOM cost	• Primary current sensing in isolated converters
Integrated CS of IPS	• Board space saving • Improved current sensing accuracy	• Integrated power stage only supports non-isolated buck topology	• Non-isolated buck converters

Shunt resistors are suitable for all applications, whereas PCB trace sensing is recommended only for high-current applications.

The XDPP1100 supports all the current sense methods listed in Table 16. The resolution of the CS ADC can be configured to be 100 µV or 1.45 mV depending on the application. When 100 µV is selected, the CS is set to high-gain mode and an internal amplifier is used. When the 1.45 mV LSB is selected, the CS is set to low-gain mode. The low-gain mode supports a reference level of either GND or 1.2 V. Typical CS should have the reference set to GND. Use the 1.2 V reference level when working with an Integrated power stage (IPS) that has integrated CS.

The multiple gain modes of XDPP1100 offer flexibility when sensing current. It could directly sense the voltage drop on a shunt resistor or sense the signal through an external amplifier. The external amplifier is recommended when the PCB layout does not allow placement of the XDPP1100 next to the CS resistor.

The XDPP1100 offers board-level trimming, which allows the user to adjust the CS gain and offset on the fly. Once the parameters are fine-tuned to achieve the desired accuracy, the configuration can be stored in the non-volatile memory in the final product. The temperature compensation is implemented in the XDPP1100 FW patch with the temperature coefficient set to 0.0039. If a different temperature coefficient applies, the user could use a FW patch and have a user-defined temperature coefficient.

DCR current sensing is usually used in low-voltage applications where the output voltage is lower than 1.5 V. This is not common in isolated power supplies, so the XDPP1100 doesn’t support DCR current sensing in isolated topologies. One exception to DCR current sensing is the inverted buck-boost topology, where the inductor can be connected to IC reference GND. In this case DCR current sensing is supported.

For the XDPP1100, the reference of the CS should at around the ground level except for the IPS mode. The IC could take common mode input voltage, a negative voltage no more than -280 mV to IC ground. This typically is more than sufficient for PCB copper current sensing.

3.2 Understanding XDPP1100 current sense

3.2.1 CS ADC

The XDPP1100 implements dedicated current ADC (IADC) to sense the input or output current. Please see chapter 1.4 for an introduction to IADC. The resolution of the current ADC is configurable by register **isenX_gain_mode**. It allows the user to configure the resolution based on application requirements. It offers two levels of resolution: a high-gain mode with 100 µV/LSB, and a low-gain mode with 1.45 mV/LSB. The input voltage range of both gain modes is listed in Table 17.

Table 17 Current ADC input voltage range

Parameters	Gain Mode	Min.	Typ.	Max.	Units	Remarks
Input differential voltage range	High gain	-22		22	mV	100 µV/LSB
Input differential voltage range	Low gain	-280		395	mV	1.45 mV/LSB

The XDPP1100 current ADC is a 9.25-bit tracking ADC. The input voltage range can be calculated by $2^{9.25} \times LSB$. The actual input voltage range is smaller than the calculated value. For exmale, the 100 µV/LSB has a -22 mV to +22 mV input voltage range, not -30.5 mV to +30.5 mV. This is because some ADC codes are allocated to do offset correction, so the valid range is less than the ideal range. The tracking ADC has dynamic step size, which automatically adjusts based on the difference between input and output. The maximum step size is 5 LSB.

The negative input indicates negative current flows through the sense resistor. The negative current sense capability is important for output current. It can be used for under-current protection, negative droop function and active current sharing.

The XDPP1100 has two input pin pairs for current sensing:

- ISEN/IREF
- BISEN/BIREF

Generally, either input pin pair can be used for both input and output current sensing. However, there are a few application-related limitations. Here are some rules for current sensing:

- In a single-loop system using VMC, either input pin pair can be used for secondary- or primary-side current sense
- In a single-loop system using PCMC, an ISEN/IREF pin pair must be connected to the current being controlled, i.e. connecting to I_{OUT} for secondary-side PCMC or connecting to I_{IN} for primary-side PCMC
- In a single-loop dual-phase system using VMC, an ISEN/IREF pin pair must be used for phase 1 secondary CS and a BISEN/BIREF pin pair must be employed for phase 2 secondary CS
- In a dual-loop system using either VMC or PCMC, an ISEN/IREF pin pair must be used for the first loop (loop 0) and a BISEN/BIREF pin pair must be employed for the second loop (loop 1)

A FB-FB converter utilizing primary-side PCMC is an example of an application where both currents I_{IN} and I_{OUT} are sensed. Typical connection of this converter is shown in **Figure 40**. The primary-side current is sensed through a current transformer and fed to the ISEN/IREF pin pair. The I_{OUT} is sensed through a shunt and fed to the BISEN/BIREF pin pair. In addition to the PCMC, the primary-side current information is used for I_{IN} telemetry and over-current fault protection. The secondary-side current information is employed for various functionalities, e.g., I_{OUT} telemetry and fault protections, droop control and current sharing.

Note: In this chapter, register names begin with ispX or ceX; X = 0 for ISEN/IREF input pins, X = 1 for BISEN/BIREF input pins.

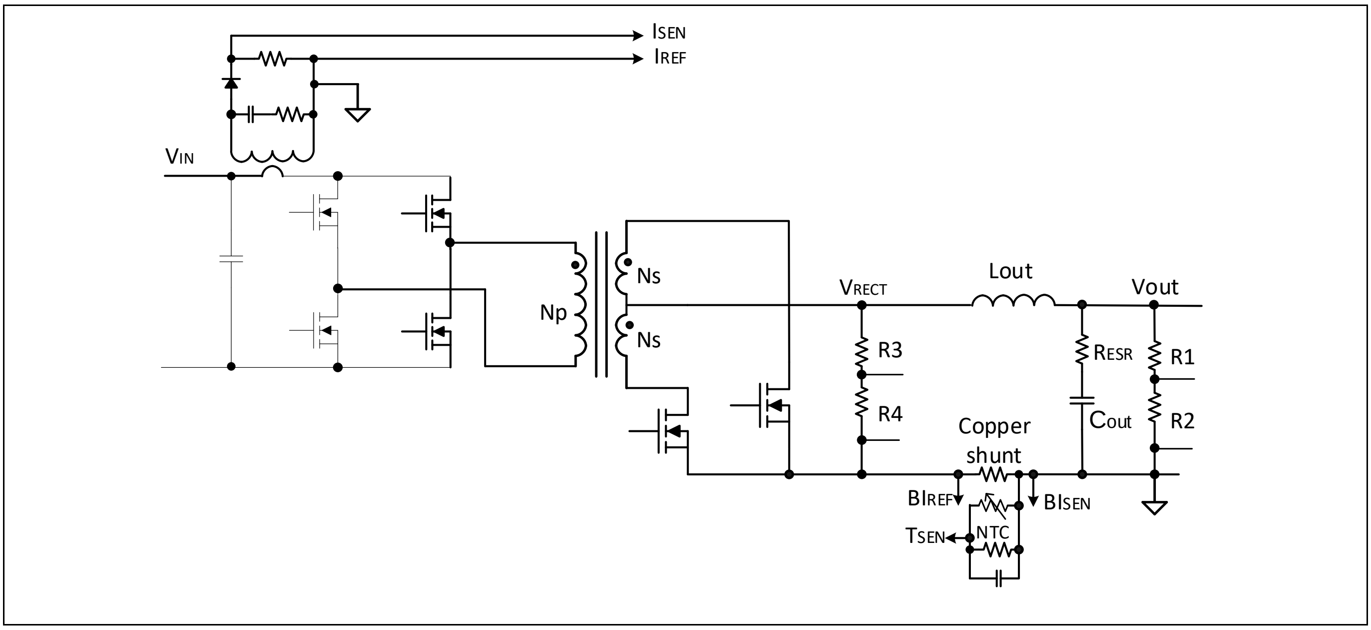


Figure 40 FB converter with primary and secondary CS

Figure 41 is an example of interleaved ACF topology. In the interleaved topologies, the secondary current of two phases makes up the total output current. Current balancing between the two phases is implemented (for details see chapter 8).

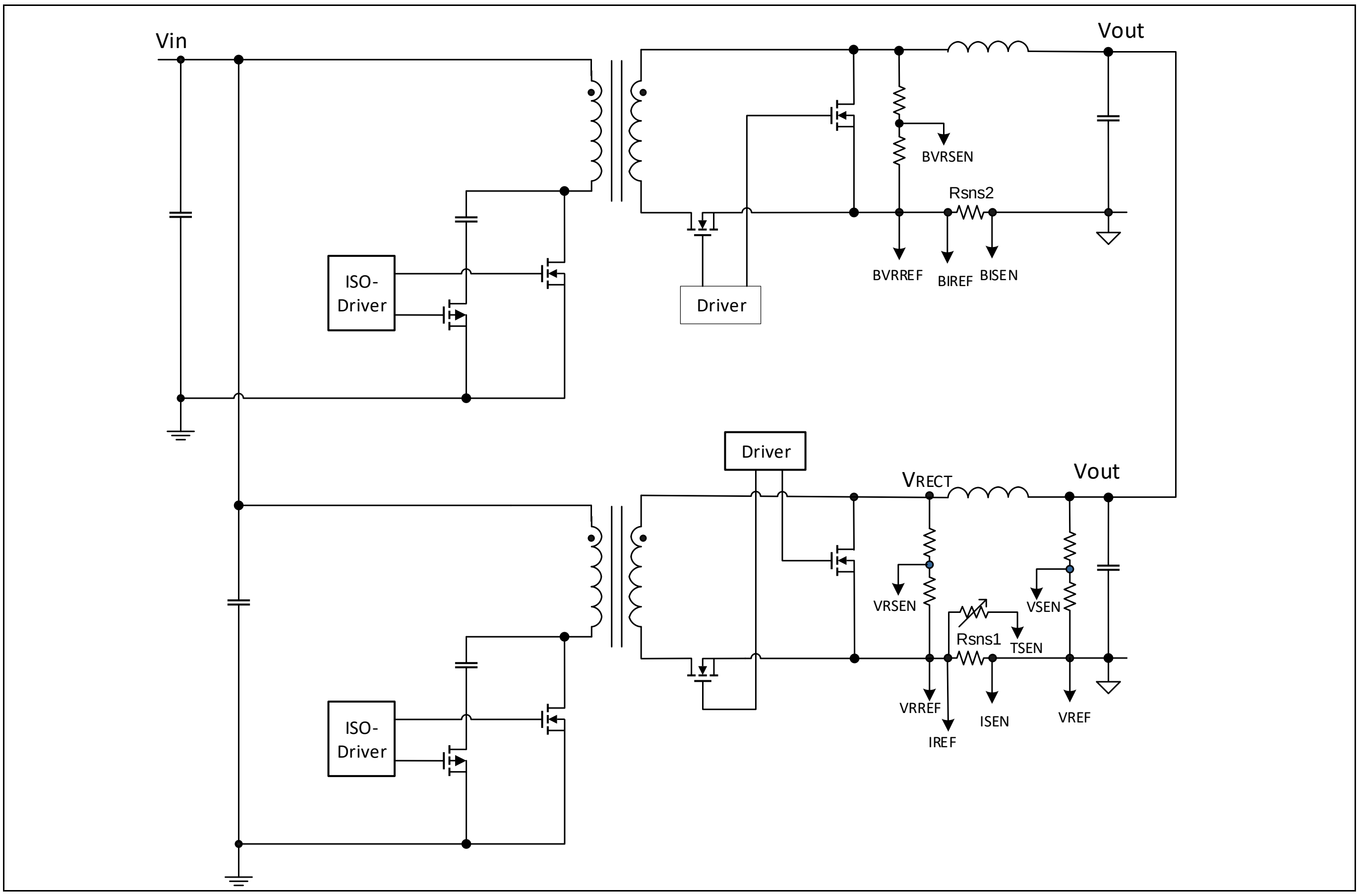


Figure 41 Interleaved ACF

3.2.2 Current estimator

The XDPP1100 current ADC operates at 25 MHz frequency. Exceptional noise immunity is achieved by the use of the internal phase current estimator. Based on the state of the PWM pulse, the controller continuously predicts each individual phase DC and ripple current. The result of the prediction is combined with the actual measured phase current to be processed by the controller. Hence, the instantaneous noise in the measurement can be filtered out without losing the valuable ripple current information. In addition, the XDPP1100 makes multiple current measurements and the readings are averaged over every switching cycle.

Figure 42 is an example of HBCT topology. Q1 and Q2 are primary switches, SR1 and SR2 are the secondary SRs. Q1 and Q2 turn-on alternately, making half of the input voltage ($V_{IN}/2$) apply to the transformer primary winding. During Q1 or Q2 turn-on time, current in the output inductor L increases. During Q1 and Q2 turn-off time, current in the output inductor decreases.

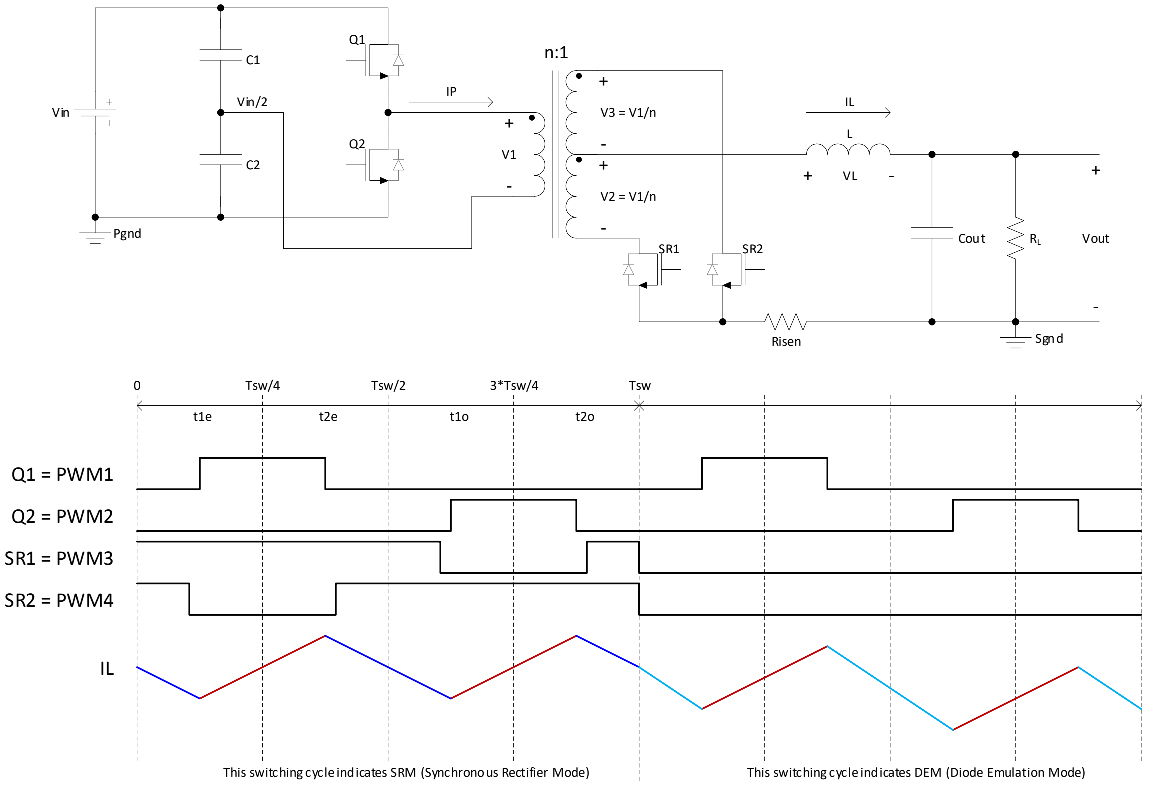
During Q1 or Q2 turn-on time, the inductor current rising slope can be calculated by:

$$didt_{on} = \frac{(\frac{V_1}{n} - V_{OUT})}{L} \tag{3.1}$$

Here, V_1 is transformer primary voltage, and n is the transformer turns ratio. In the rest of the document V_1/n is called V_{RECT}, which represents the input voltage.

$V_1 = \frac{V_{IN}}{2}$, for HB topology.

$V_1 = V_{IN}$, for FB, forward and buck topologies.



ON = (Q1 or Q2) ➔ ce0_on_mask0[0] = 1, other ce0_on_mask0[] = 0, ce0_on_mask1[1] = 1, other ce0_on_mask1[] = 0

OFF = not(ON) and (SR2 or SR1) ➔ ce0_off_mask0[3] = 1, other ce0_off_mask0[] = 0, ce0_off_mask1[2] = 1, other ce0_off_mask1[] = 0

HIZ = not(ON) and not(OFF)

Figure 42 Output inductor current

The inductor current falling slope can be calculated based on two different cases. Case 1 is when the SR is enabled (dark blue IL trace in **Figure 42**); case 2 is when the SR is disabled and the secondary current flows through the body diode of the SR MOSFET (DE mode, light blue IL trace in **Figure 42**).

$$didt_{off\_SR} = \frac{(-V_{OUT})}{L} \tag{3.2}$$

$$didt_{off\_DE} = \frac{(-V_{OUT}-V_F)}{L} \tag{3.3}$$

V_F is the body diode voltage drop.

To calculate the current slope, the XDPP1100 needs to know the input voltage, the output voltage, transformer turns ratio, the output inductance, and the body diode voltage drop. The input and output voltage are continuously measured through the VADCs. The user has the option to configure which VADC is used for V_{IN} sense (**ceX_vrect_sel**) and which ADC is used for V_{OUT} sense (**ceX_vout_sel**). The transformer turns ratio is configured by PMBus command MFR_TRANSFORMER_SCALE. The output inductance information is configured by register **ceX_kslope_didv**. The body diode voltage drop is fixed to 0.5 V in the chip.

With all the parameters, the XDPP1100 current estimator could predict the output inductor current level and ripple. To utilize the current estimator, it is recommend to sense output current before the output capacitor to include ripple current information (on the left side of C_{OUT} in **Figure 42**). This is the inductor current. With the inductor current sensing, the XDPP1100 can implement secondary PCMC, and cycle-by-cycle PCL.

A similar approach works for primary current sensing. Additional information to that required for primary current sense is the transformer magnetizing inductance, which is configured by the register **ceX_kslope_lm**.

Figure 43 is the block diagram of the XDPP1100 CS ADC and current estimator.

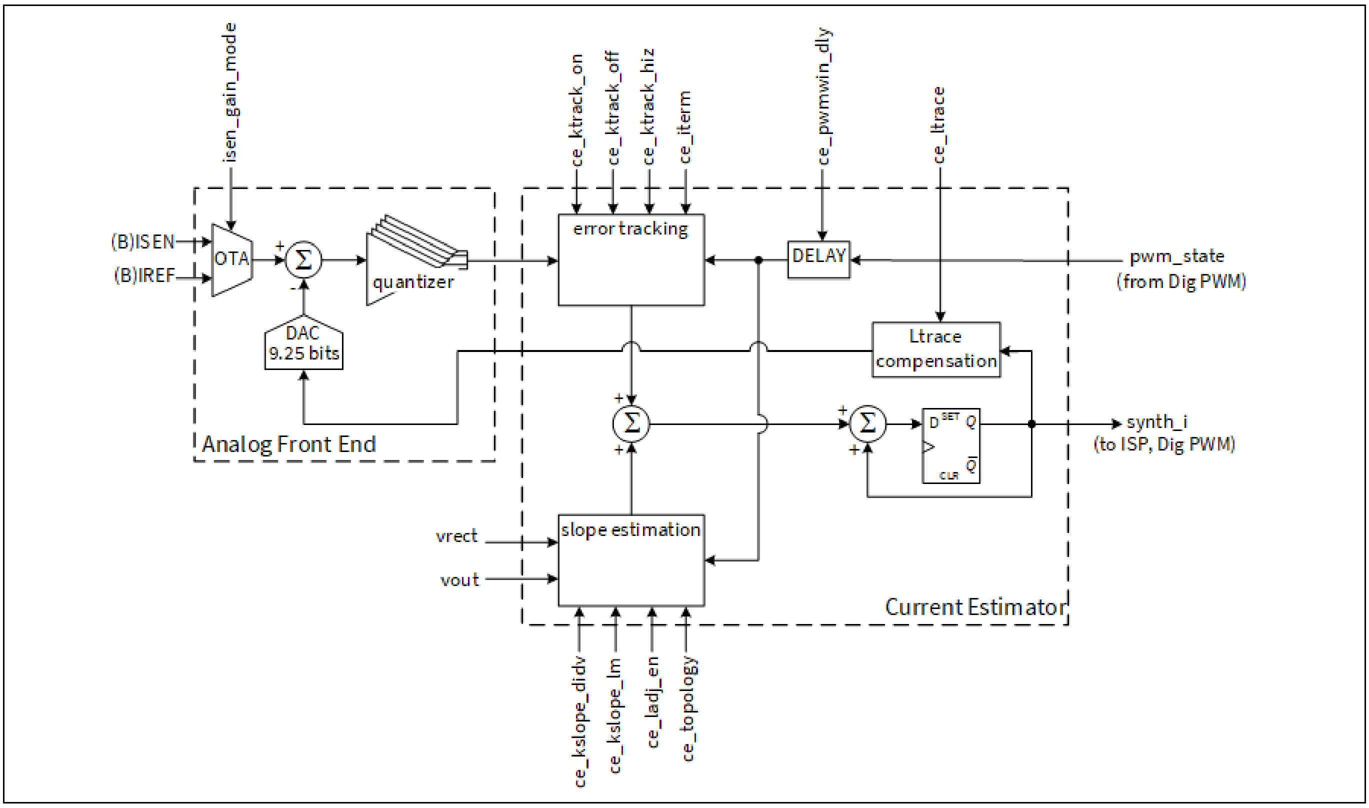


Figure 43 The XDPP1100 current sense

The current estimator has four main functions:

- PWM state delay function
- Slope estimator
- Error tracking function
- Parasitic inductance L_{trace} compensation

3.2.2.1 PWM state

The PWM has three states with respect to the inductor current cycle for a buck-derived isolated topology, as illustrated in **Figure 44**. These states are:

- On-state: inductor current rising slope when the PWM FET is on
- Off-state: inductor current falling slope when the PWM FET is off and the SR FET is on
- HiZ (high impedance) state: inductor current slope when both switches are off; in the case of positive current, the slope is falling toward zero, and for negative current the slope is increasing toward zero

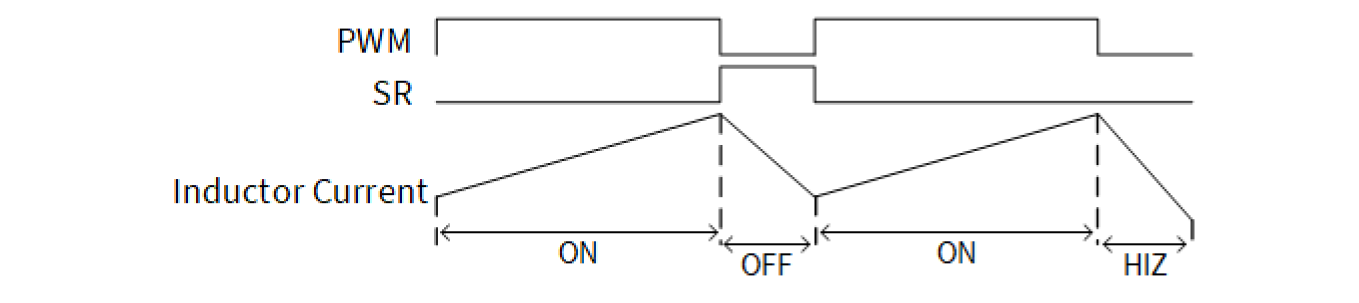


Figure 44 PWM state definition with respect to the inductor current cycle

In the actual system, there is delay between the internal PWM state and the actual PWM state observed from the output inductor. In order to better align the internal state to the sensed current waveform, the internal PWM can be delayed, which is possible via register **ceX_pwmwin_dly**. While adjusting this parameter, all possible delay sources (e.g., controller output latency, isolator latency, driver latency) within the system should be considered. Delay time is defined by (**ceX_pwmwin_dly** + 1) * 10 ns. It should be noted that an exact unit-by-unit delay match is not required, as the error tracking corrects minor timing mis-matches at the PWM state transitions.

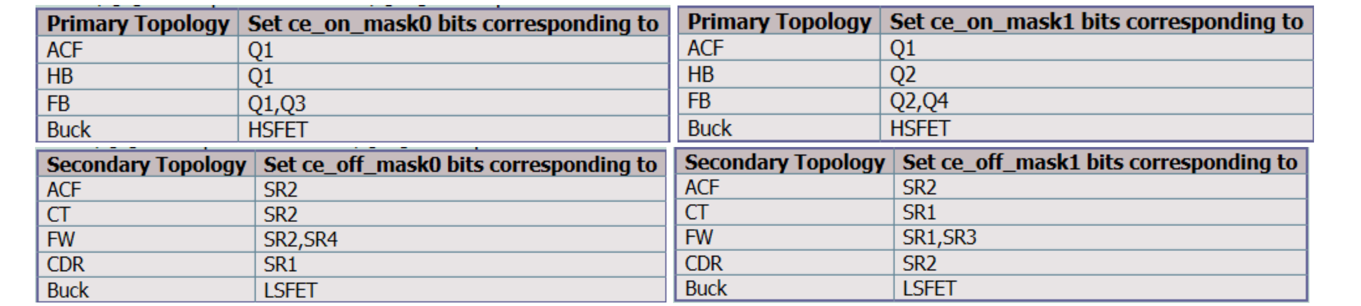
Depending on the application, there are a different number of FETs that define the on- and off-times in a certain topology. Each FET is controlled by PWM signal, and for current reconstruction purposes, the FETs contributing to the PWM state are relevant for the state programming. The PWM states are programmed for ISEN and BISEN input pins via the following registers:

- For defining ISEN on-time: **ce0_on_mask0[11:0]** and **ce0_on_mask1[11:0]**
- For defining ISEN off-time: **ce0_off_mask0[11:0]** and **ce0_off_mask1[11:0]**
- For defining BISEN on-time: **ce1_on_mask0[11:0]** and **ce1_on_mask1[11:0]**
- For defining BISEN off-time: **ce1_off_mask0[11:0]** and **ce1_off_mask1[11:0]**

Each register consists of 12 bits, where each bit field corresponds to a specific PWM output. For instance, in the above registers, bit [n] corresponds to the pin PWM n+1 so that:

- Bit [0] corresponds to pin PWM1
- Bit [11] corresponds to pin PWM12

The bit corresponding to the specific PWM output that drives a FET that defines the on- or off-time needs to be set to 1 in the relevant register. In bridge topologies there are two on-/off-states per switching cycle; therefore, mask0 defines the first on-/off-state and mask1 the second. In non-bridge topologies (e.g., ACF, buck) mask1 is typically set to the same value as mask0.



Primary Topology	Set ce_on_mask0 bits corresponding to
ACF	Q1
HB	Q1
FB	Q1,Q3
Buck	HSFET

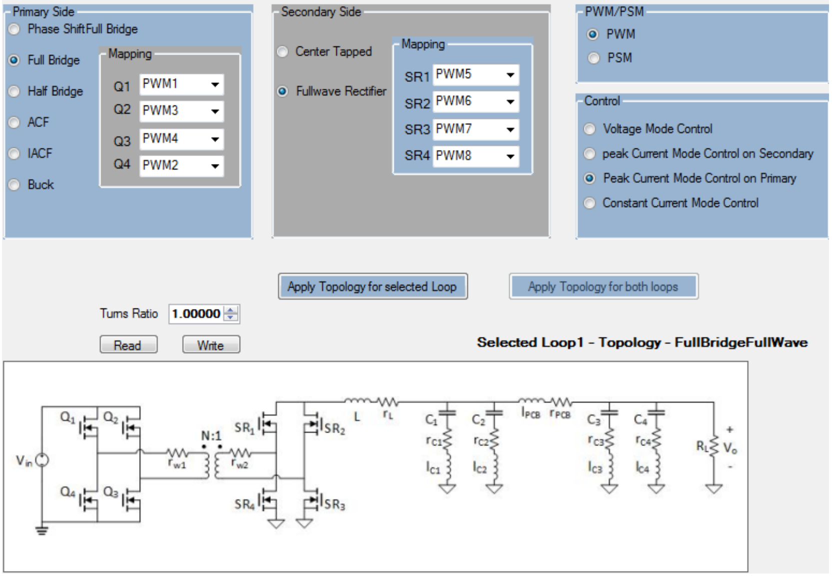
Primary Topology	Set ce_on_mask1 bits corresponding to
ACF	Q1
HB	Q2
FB	Q2,Q4
Buck	HSFET

Secondary Topology	Set ce_off_mask0 bits corresponding to
ACF	SR2
CT	SR2
FW	SR2,SR4
CDR	SR1
Buck	LSFET

Secondary Topology	Set ce_off_mask1 bits corresponding to
ACF	SR2
CT	SR1
FW	SR1,SR3
CDR	SR2
Buck	LSFET

Figure 45 ce_on_mask and ce_off_mask definition

The ceX_on_mask and ceX_off_mask registers will be configured by the GUI design tool in the "Device Topology" section (**Figure 46**) based on topology selection and PWM mapping. It is not recommended to change the value manually.



Primary Side
- ○ Phase ShiftFull Bridge
- ● Full Bridge
- ○ Half Bridge
- ○ ACF
- ○ IACF
- ○ Buck

Mapping
- Q1 PWM1
- Q2 PWM3
- Q3 PWM4
- Q4 PWM2

Secondary Side
- ○ Center Tapped
- ● Fullwave Rectifier

Mapping
- SR1 PWM5
- SR2 PWM6
- SR3 PWM7
- SR4 PWM8

PWM/PSM
- ● PWM
- ○ PSM

Control
- ○ Voltage Mode Control
- ○ peak Current Mode Control on Secondary
- ● Peak Current Mode Control on Primary
- ○ Constant Current Mode Control

Turns Ratio 1.00000

Selected Loop1 - Topology - FullBridgeFullWave

Figure 46 FBFW PWM mapping

Take the FBFW with secondary current sensed by ISEN as an example (**Figure 46**). The GUI would assign the following values to the ceX_on_mask and ceX_off_mask registers.

ce0_on_mask0 is determined by Q1, Q3. With Q1=PWM1 (bit [0]), Q3=PWM4 (bit [3]), the **ce0_on_mask0**=$2^0+2^3=1+8=9$

ce0_on_mask1 is determined by Q2, Q4. With Q2=PWM2 (bit [1]), Q4=PWM3 (bit [2]), the **ce0_on_mask1**=$2^1+2^2=2+4=6$

ce0_off_mask0 is determined by SR2, SR4. With SR2=PWM6 (bit [5]), SR4=PWM8 (bit [7]), the ce0_off_mask0=$2^5+2^7=32+128=160$

ce0_off_mask1 is determined by SR1, SR3. With SR1=PWM5 (bit [4]), SR3=PWM7 (bit [6]), the ce0_off_mask1=$2^4+2^6=16+64=80$

The ce1 on/off mask of the dual-loop application is also handled by the Device Topology tool by the second loop.

3.2.2.2 Slope estimator

The normalized current slope is defined by the parameter **ceX_kslope_didv**. The definition for the current slope depends on whether the current is sensed on the primary or secondary side, and on the topology.

The primary- and secondary-side current waveforms are different, as illustrated in **Figure 47**. The continuous secondary-side current can be sensed during any PWM state. However, the primary-side current can be sensed only while PWM is in the on-state. This means that in the off- and HiZ states tracking through the ADC is not possible. Therefore, the tracking should be disabled by setting both register values, **ce_ktrack_off** and **ce_ktrack_hiz**, to zero for primary-side current.

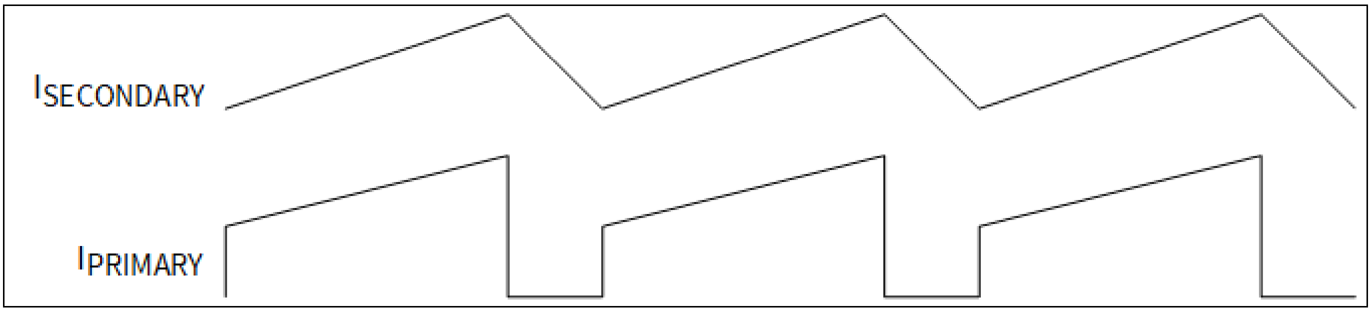


Figure 47 Primary-side and secondary-side current waveforms

Taking the FB-FB PCMC control as an example, we have ISEN (ce0) measuring primary current, while BISEN (ce1) measures secondary current.

When the current is sensed on the primary side, the **ceX_kslope_didv** calculation must take the transformer turns ratio (N_P/N_S) into consideration:

$$ce0\_kslope\_didv = \frac{1V \cdot 10ns}{Lout(H) \cdot APC \cdot \frac{Np}{Ns}} \times 2^{13} \tag{3.4}$$

L_{OUT} is the output inductance in Henry and APC is the “amps per code” in amps, as defined by PMBus command MFR_IOUT_APC. The LSB of **ce_kslope_didv** is 2^{-13}.

For output current sense, the equation changes to:

$$ce1\_kslope\_didv = \frac{1V \cdot 10ns}{Lout(H) \cdot APC} \times 2^{13} \tag{3.5}$$

Please note that for the FB-FB primary PCMC topology, even though it is a single-loop application, it is required to use the 40-pin version XDPP1100-K040 to enable both primary and secondary CS. The primary-side current should be sensed by ISEN/IREF input and configured by ce0 and PMBus page 0 MFR_IOUT_APC. The secondary-side current should be sensed by BISEN/BIREF input and configured by ce1 and PMBus page 1 MFR_IOUT_APC.

The calculation of MFR_IOUT_APC is given by the equation:

$$MFR\_IOUT\_APC \;=\; ISEN\_LSB/Rsns$$

The ISEN_LSB is the resolution of CS ADC which is determined by the **isenX_gain_mode** register. XDPP1100 offers two levels of gain: 100 µV and 1.45 mV, which reference to ground (GND); and one IPS mode, which has a resolution of 1.45 mV and references to a DC bias range from 1.11 V to 1.6 V. The ISEN_LSB is 1.45 mV when **isenX_gain_mode** =2.

The Rsns is the equivalent current sense resistor value. For example, the secondary current is sensed by PCB copper trace (0.15 mΩ). The signal is amplified by an external op-amp with a gain of 201. The signal is then divided by resistor divider at the input of XDPP1100 with a ratio of 0.1423. The equivalent secondary sense resistor is 0.15 x 201 x 0.1423 = 4.29 mΩ. With **isenX_gain_mode** =2, the secondary MFR_IOUT_APC = 1.45 mV/4.29 mΩ = 0.338 A.

Here is a design example of a FB-FB primary PCMC converter. With the following system settings, we could calculate **ceX_kslope_didv**:

Output inductor L_{OUT} = 0.4 µH; loop 0 MFR_IOUT_APC = 0.0996 A (primary CS gain); loop 1 MFR_IOUT_APC = 0.35 A (output CS gain); N_P/N_S = 3.

$$ce0\_kslope\_didv = \frac{1V \cdot 10ns}{0.4 \cdot 10^{-6}(H) \cdot 0.0996 \cdot \frac{3}{1}} \times 2^{13} = 685 \qquad (3.6)$$

$$ce1\_kslope\_didv = \frac{1V \cdot 10ns}{0.4 \cdot 10^{-6}(H) \cdot 0.35} \times 2^{13} = 586 \qquad (3.7)$$

In addition to the **ceX_kslope_didv**, the primary CS also needs to configure the **ceX_kslope_lm** register. **ceX_kslope_lm** defines the primary magnetizing current slope. In this example, primary current is sensed by ISEN (ce0), and the transformer magnetizing inductance is 25 µH.

$$ce0\_kslope\_lm = \frac{\frac{Np}{Ns} \cdot 1V \cdot 10ns}{Lm(H) \cdot APC} \times 2^{13} \qquad (3.8)$$

$$ce0\_kslope\_lm = \frac{\frac{3}{1} \cdot 1V \cdot 10ns}{25 \cdot 10^{-6}(H) \cdot 0.0996} \times 2^{13} = 98 \qquad (3.9)$$

The secondary CS does not need to configure the **ceX_kslope_lm** register and should be set to 0.

The register **ceX_ps_current_emu** needs to be set to 1 for primary-side CS. In this example, ISEN (ce0) is used for primary CS, thus **ce0_ps_current_emu** should be set to 1. When a single-loop topology requires the use of both ISEN and BISEN ADCs, PMBus command FW_CONFIG_REGULATION should set on bit [1] EN_PRIM_ISENSE to "1" in order to enable both the ISEN and BISEN current sense paths simultaneously. Single-loop dual-phase or dual-loop topologies don't require being set to EN_PRIM_ISENSE.

In the real application the output inductance has a certain tolerance, and therefore, its contribution to **ceX_kslope_didv** will vary. This tolerance can be corrected by:

- The error tracking part of the current estimator
- Setting parameter **ceX_ladj_en** to "1", which enables autocorrect for errors in **ceX_kslope_didv** based on the incoming CS waveform

In addition to the inductance tolerance, the inductor value may vary with the respect to the current. This variation is typically non-linear and XDPP1100 provides a linear correction, as illustrated in **Figure 48**. Register **ceX_dt_l_slope** defines the L slope dependence on the current. As illustrated in the figure, the compensation is specified by:

- First point: use inductance value at 0 A
- Second point: user definable based on the best-fit curve for each application

Therefore, assuming L_0 is the inductance at zero current, the parameter **ceX_dt_l_slope** is computed according to Equation (3.10).

$$ceX\_dt\_I\_slope \; = INT\left(2^{14} \times \left(\left(\frac{L_0}{L_1}\right) - 1\right) \times \frac{APC}{I_1}\right) \qquad (3.10)$$

Where:

- L_0 = inductance at 0 A
- L_1 = inductance at user-selected I_1
- APC = amps per code from MFR_IOUT_APC

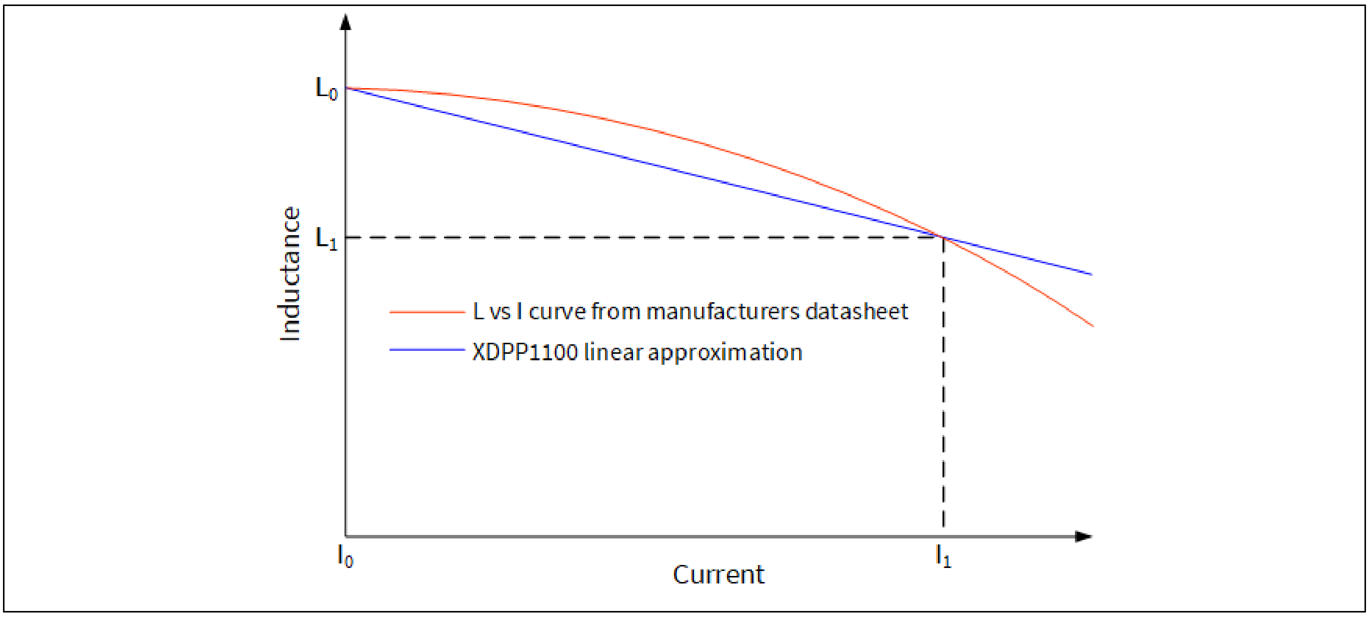


Figure 48 **Inductance variation with the current**

The voltage across the inductor is obtained based on the measured voltages: V_{OUT}, V_{IN} and/or V_{RECT}, depending on the topology and the PWM state. The XDPP1100 controller supports inductor voltage equations for buck-derived, as well as boost and buck-boost topologies. The topology is selected via the register **ceX_topology**. The supported topologies and their corresponding inductor voltage equations for different PWM states are shown in Table 18.

Table 18 **Topology selection for current estimator**

ce_topology	Topologies	V_{ON} equation	V_{OFF} equation	V_{HiZ} equation
0	Buck, ACF, HB, FB	V_{RECT} - V_{OUT}	-V_{OUT}	-V_{OUT} - V_{body}
1	Boost	V_{RECT}	V_{RECT} - V_{OUT}	V_{RECT} - V_{OUT} - V_{body}
2, 3	Buck-boost	V_{RECT}	-V_{OUT}	-V_{OUT} - V_{body}

The voltage V_{body} refers to the voltage across the body diode, and it is hard-coded to 0.5 V. The measured voltages V_{OUT} and V_{RECT} can be selected via the following registers:

- **ceX_vout_sel** for V_{OUT} selection
- **ceX_vrect_sel** for V_{RECT} selection

Table 19 and Table 20 provide the different voltage source options for V_{OUT} and V_{RECT}, where X = 0 refers to ISEN/IREF input pins and 1 refers to BISEN/BIREF. The CDR in Table 19 indicates current doubler topology on the secondary.

Table 19 **Current estimator V_{OUT} source**

ceX_vout_sel	CE0 (ISEN) associated V_{OUT}	CE1 (BISEN) associated V_{OUT}
0	Loop 0 V_{OUT}	Loop 0 V_{OUT}
1	Loop 0 V_{OUT} CDR	Loop 0 V_{OUT} CDR

ceX_vout_sel	CE0 (ISEN) associated V_{OUT}	CE1 (BISEN) associated V_{OUT}
2	Loop 0 V_{OUT}	Loop 1 V_{OUT}
3	Loop 0 V_{OUT} CDR	Loop 1 V_{OUT} CDR

Table 20 Current estimator V_{RECT} source

ceX_vrect_sel	CE0 (ISEN) associated V_{RECT}	CE1 (BISEN) associated V_{RECT}
0	VS1 (VRSEN/VRREF)	VS1 (VRSEN/VRREF)
1	VS2 (BVRSEN/BVRREF)	VS2 (BVRSEN/BVRREF)
2	pid_ff_vrect_override	pid_ff_vrect_override
3	PRISEN	PRISEN
4	VS1 (VRSEN/VRREF)	Loop 0 V_{OUT}
5	VS2 (BVRSEN/BVRREF)	Loop 0 V_{OUT}
6	pid_ff_vrect_override	Loop 0 V_{OUT}
7	PRISEN	Loop 0 V_{OUT}

In a dual-loop system:

- The ISEN/IREF pin pair must be used to sense the current for loop 0, whereas the BISEN/BIREF pin pair can be assigned for loop 1, as shown in Table 19
- Loop 0 V_{OUT} is allowed to be used as input voltage for loop 1, as shown in Table 20.

Boost and buck-boost topology current sense

For applications other than buck-derived, proper topology for the current estimator should be selected according to Table 18. The **ceX_topology** should be set to boost or buck-boost. In case of boost topology, the inductor current can be measured only during the on-state. Therefore, the tracking must be disabled during off- and HiZ states. This can be set by selecting:

- **ceX_ktrack_off** = 0
- **ceX_ktrack_hiz** = 0

3.2.2.3 Error tracking

The error tracking block is essentially a gain block with independent gains for the on-, off- and HiZ states, as illustrated in **Figure 43**.

- Tracking ADC feedback is provided from the analog front end (AFE) quantizer output through the error tracking function
- Error tracking output is added to the slope estimator output and fed back to the AFE DAC

The tracking gain defines the relative strength of the IADC-based correction term, which is applied to the emulated current waveform. The CS tracking gains can be defined through registers:

- **ceX_ktrack_on**, for on-state
- **ceX_ktrack_off**, for off-state
- **ceX_ktrack_hiz**, for high impedance state

The value of k_{track} gain can be set in a range of 0 to 15. When set to 0, the current estimator will not track to ADC sensed current but will fully rely on estimation, which is calculated based on voltage and inductor value. When k_{track} gain is set to 15, the current estimator uses the actual sensed current. Other values set the weight of sensed current over estimated current with a ratio of ceX_ktrack_xxx/16.

In this block, register **ceX_iterm** defines an integral coefficient, which is applied across all states if set to a non-zero value. Setting this CS tracking error integrator coefficient to 0 disables the error accumulation.

The PWM transition might cause noise on the CS signal. Therefore, certain parameters exist that are used to blank the ADC feedback. The following registers can be used to define the transition time windows:

- **ceX_blank_ne_dly** defines the time subsequent to the PWM on-state negative edge, during which the error tracking output is blanked to 0 (i.e. the gain is set to 0)
- **ceX_blank_pe_dly** defines the time after a PWM on-state positive edge, during which the error tracking output is blanked to 0

The LSB of the blanking time is 40 ns. Blanking time could be set between 0 ns and 280 ns.

3.2.2.4 Trace inductance of the PCB current sensing

The PCB trace resistance can be used for output current sensing. In the ideal case the parasitic inductance is zero and the CS voltage shape is a clear sawtooth waveform following exactly the current shape. However, in practice this trace inductance is not zero and it introduces a step to the CS waveform, as indicated by Equation (3.11):

$$V_{SEN} = R_{PCB} \times I + L_{PCB} \times \frac{dI}{dt}, \qquad (3.11)$$

Where $\frac{dI}{dt}$ is a function of V_{RECT}, V_{OUT} and L_{OUT}. **Figure 49** shows the CS voltage waveforms for the theoretical case L = 0 (V_{RPCB} in red) and L ≠ 0 (V_{SEN} in blue).

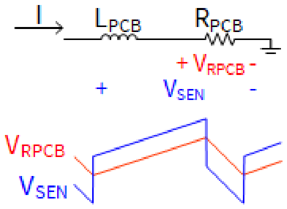


Figure 49 PCB trace inductance

Register **ce_ltrace** is provided to compensate for this step in the input to the AFE DAC. The compensation value is obtained according to Equation (3.10), where the parameter ce_ltrace_real is defined in Equation (3.12).

$$ce\_ltrace\_real = \frac{L_{PCB}(H)}{(2 \times R_{PCB}(\Omega) \times 10e{-}8sec)}, \qquad (3.12)$$

The calculation could be done by GUI design tool, by providing the value of L_{trace} and R_{trace} (**Figure 50**).

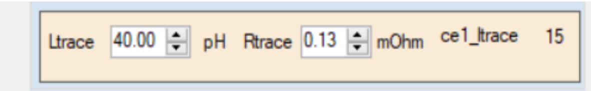

Ltrace 40.00 pH Rtrace 0.13 mOhm ce1_ltrace 15

Figure 50 GUI design tool – CS configuration

3.2.2.5 Bypass the current estimator

The purpose of the current estimator is to increase CS accuracy and improve noise immunity by predicting the current ripple in the inductor or transformer combined with the actual measured current. For this reason, it is recommended to sense the inductor current and utilize the current estimator. For example, place the CS shunt before the output filter capacitor when sensing output current.

If the user wants to sense the DC output current, it is also possible to bypass the current estimator. Setting the **ce_kslope_didv** to 1 will minimize the slope estimation and the current sense relies on the IADC measurement.

3.2.3 Current telemetry

3.2.3.1 Output current telemetry

Figure 51 shows the output current telemetry block diagram.

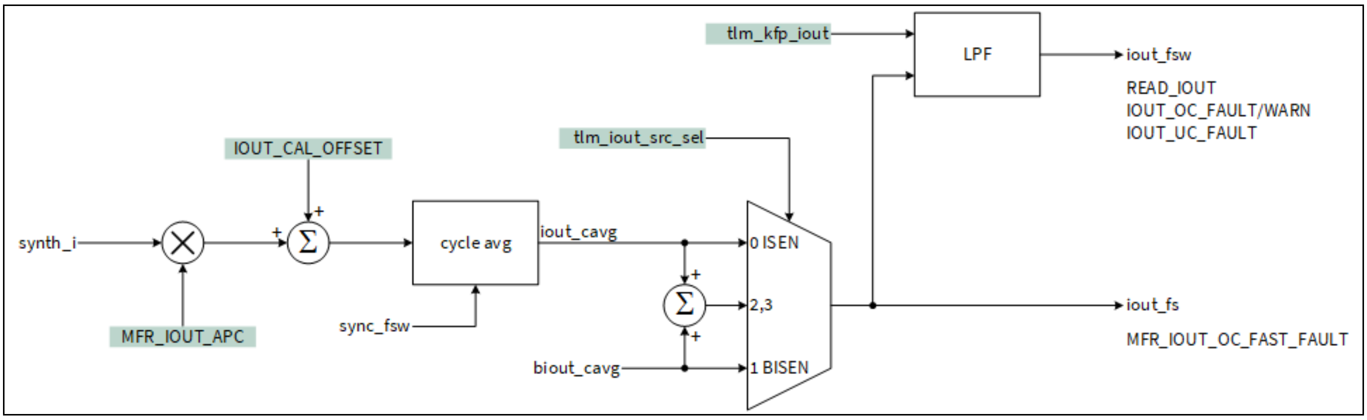


Figure 51 Output current telemetry

- Current estimator output synth_i (represented in ADC codes) is multiplied by the PMBus command MFR_IOUT_APC (APC refers to “amps per code”, converting the code-based word synth_i into a digital representation of amps).
- The resulting APC multiplication output is then added to the PMBus command IOUT_CAL_OFFSET to adjust for offset errors.
- The reconstructed waveform in amps is averaged over one switching cycle to produce a cycle average current.

The averaged current is further processed downstream in the telemetry and fault blocks. The I_{OUT} telemetry block has two output signals: iout_fsw is the low-pass filtered output current, and the iout_fs is unfiltered output current. The LPF BW is configured by register **tlmX_kfp_iout**. The LPF can be bypassed by setting the **tlmX_kfp_iout** register to 63.

The low-pass filtered I_{OUT} is used for telemetry and normal over-current and under-current fault protection. The unfiltered output is used for fast over-current fault protection.

The output current source should be selected by register **tlm_iout_src_sel**. See the description in Table 3.

Table 21 tlm_iout_src_sel configuration table

Value	Output current source
0	ISEN

Value	Output current source
1	BISEN
2 or 3	Sum of ISEN and BISEN inputs (dual-phase)

3.2.3.2 Input current telemetry

Figure 52 shows the input current telemetry block diagram. Similar to the output current telemetry, when the input current is measured by ISEN or BISEN, the synthesis current from the current emulator is multiplied by the PMBus command MFR_IOUT_APC, then summed with the PMBus command IOUT_CAL_OFFSET. The waveform is averaged over one switching cycle and filtered by **tlm_kfp_iin** LPF and sent to the telemetry and fault protection blocks.

Note that the PMBus Page 0 commands for MFR_IOUT_APC and IOUT_CAL_OFFSET correspond to the ISEN path and the PMBus Page 1 commands correspond to the BISEN path.

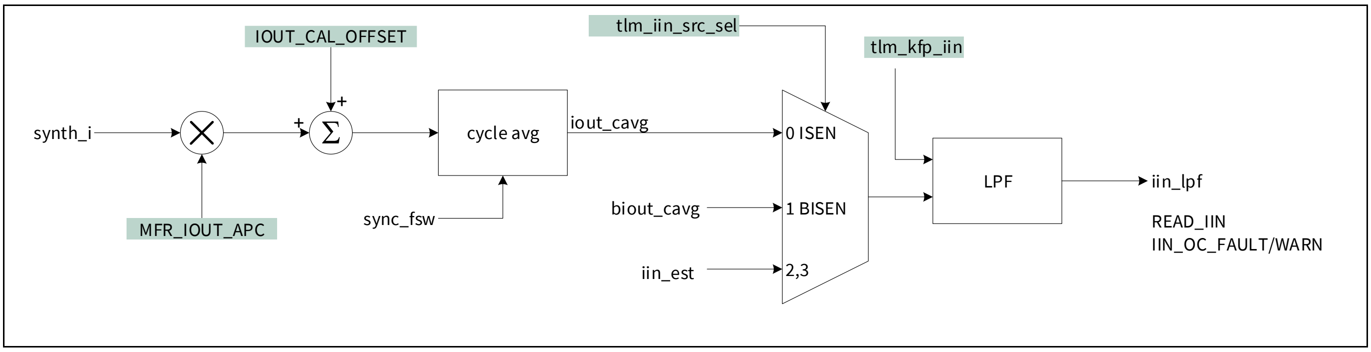


Figure 52 Input current telemetry

The input current source should be selected by register **tlm_iin_src_sel**.

Table 22 tlm_iin_src_sel configuration table

Value	Input current source
0	ISEN
1	BISEN
2 or 3	Estimated input current based on I_{OUT}

The estimated input current is calculated by the following equation:

$$Iin_{est} = I_{OUT} \times [(1 - alpha) \times \frac{V_{OUT}}{V_{IN}} + alpha \times Duty \times tlm\_transformer\_scale\_loop] \qquad (3.13)$$

The alpha in the equation is defined by register **tlm_in_est_alpha**. The tlm_transformer_scale_loop is defined by PMBus command MFR_TRANSFORMER_SCALE and converter topology. The tlm_transformer_scale_loop is managed by FW and does not need to be configured by the user.

3.2.3.3 Trim current telemetry

The PCB copper shunt, typically in the sub-mΩ range, varies from board to board as well as over temperature range. A board-level trim is required for accurate current telemetry and protections. This can be done by trimming the gain (MFR_IOUT_APC) and offset (IOUT_CAL_OFFSET) of the CS. Please note that even though the command name contains IOUT, it trims the IIN gain and offset if the ADC is assigned for input current sense.

- MFR_IOUT_APC and IOUT_CAL_OFFSET of loop 0 configures ISEN/IREF

- MFR_IOUT_APC and IOUT_CAL_OFFSET of loop 1 configures BISEN/BIREF

The temperature compensation is also required to eliminate the copper resistance drift over temperature. The compensation can be enabled by setting bit [9] of PMBus command FW_CONFIG_REGULATION. The default temperature coefficient is 0.00393. If a different temperature coefficient is preferred, the user can override the 0.00393 with a FW patch. To get accurate compensation, the temperature sensor must be placed very close to the shunt resistor. The FW takes the measurement from READ_TEMPERATURE_1 for temperature compensation. The user has the option to configure which temperature source should be mapped to READ_TEMPERATURE_1.

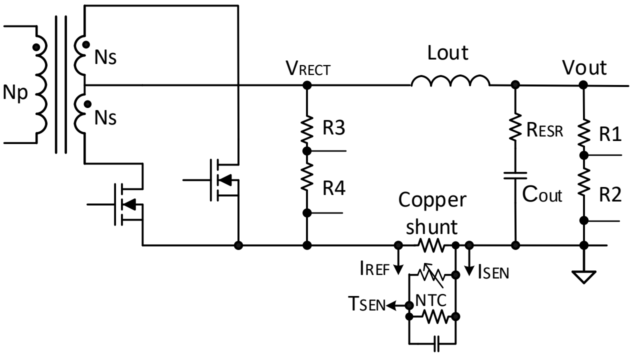


Figure 53 **Output current sense through PCB trace with temperature compensation**

3.2.4 Fault protections

The main purpose of the CS processor (ISP) is to convert digitally reconstructed current waveforms from ADC codes into a digital representation of amps. A simplified block diagram of the ISP is shown in **Figure 54**. The CS processor also includes several phase current-based fault checks as indicated in the figure. These fault checks include:

- PCL
- Negative current limit
- Short-circuit fault
- Error tracking fault

The above fault protections are part of the common fault management of the XDPP1100. The details of common fault protection are described in chapter 12.9.2.

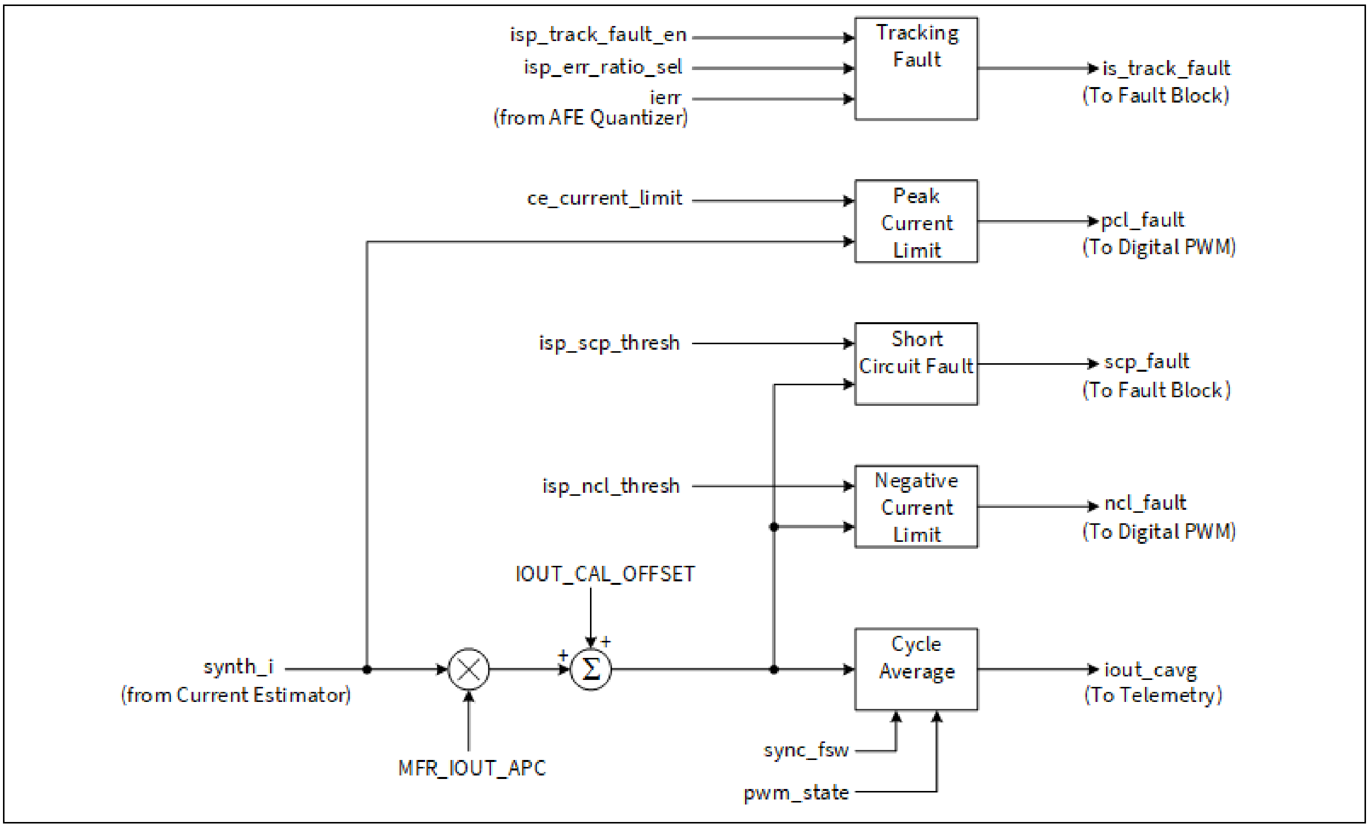


Figure 54 Simplified block diagram of the CS processor

3.3 CS register descriptions

Table 23 describes the registers used by the current sense function.

Table 23 CS relevant register descriptions

Name	Address (loop 0/1)	Bits	Description
Analog peripheral			
isen1_gain_mode	7000_0400$_H$	[13:12]	ISEN1 (ISEN) gain mode select. Defines LSB weight of ISEN1 ADC. Also defines expected reference voltage level on IREF1 (IREF). 0 = reserved 1 = 100 µV, reference level GND 2 = 1.45 mV, reference level GND 3 = 1.45 mV, reference level 1.2 V
Isen2_gain_mode	7000_0400$_H$	[15:14]	ISEN2 (BISEN) gain mode select. Defines LSB weight of ISEN2 ADC. Also defines expected reference voltage level on IREF2 (BIREF). 0 = reserved 1 = 100 µV, reference level GND 2 = 1.45 mV, reference level GND 3 = 1.45 mV, reference level 1.2 V
ISEN peripheral			

Name	Address (loop 0/1)	Bits	Description
ce_ktrack_hiz	7000_2400_H (ISEN) 7000_2800_H (BISEN)	[3:0]	CS tracking gain in the HiZ state. The tracking gain defines the relative strength of the IS ADC-based correction term applied to the emulated current waveform. The HiZ state generally refers to the state when all primary and secondary switches are off. Note that this register should be set to 0 for primary-side CS and for the boost topology. LSB = 1/16 V/V, range = 0 to 15/16 V/V
ce_ktrack_off	7000_2400_H (ISEN) 7000_2800_H (BISEN)	[7:4]	CS tracking gain in the off-state. The tracking gain defines the relative strength of the IS ADC-based correction term applied to the emulated current waveform. The off-state generally refers to the state when the output inductor current is in its downward slope cycle (e.g., primary FETs off, secondary FETs on in a bridge topology). Note that this register should be set to 0 for primary-side CS and for the boost topology. LSB = 1/16 V/V, range = 0 to 15/16 V/V
ce_ktrack_on	7000_2400_H (ISEN) 7000_2800_H (BISEN)	[11:8]	CS tracking gain in the on-state. The tracking gain defines the relative strength of the IS ADC-based correction term applied to the emulated current waveform. The on-state generally refers to the state when the output inductor current is in its upward slope cycle (e.g., primary FETs on in a bridge topology). LSB = 1/16 V/V, range = 0 to 15/16 V/V

ce_kslope_didv	7000_2400_H
	(ISEN)
	7000_2800_H
	(BISEN)
		[22:12]	Output inductor (L_{OUT}) CS slope normalized to code/samples at 1.0 V.
			For secondary CS:
			ce_kslope_didv = 1.0 V * 10 ns/(L_{OUT}(nH) * APC(A))
			For primary CS:
			ce_kslope_didv = 1.0 V * 10 ns/(N_{turn} * L_{OUT}(nH) * APC(A))
			LSB = 2^-13 V/V, range = 0.0 to 0.25 V/V

ce_pwmwin_dly	7000_2400_H
	(ISEN)
	7000_2800_H
	(BISEN)
		[27:24]	Defines delay used to align internal PWM signals to incoming CS waveform.
			Delay = (ce_pwmwin_dly + 1) * 10 ns
			LSB = 10 ns, range = 10 to 320 ns

ce_ladj_en	7000_2400_H (ISEN) 7000_2800_H (BISEN)	[30]	Inductor slope correction function enable. When enabled, the controller will attempt to auto-correct for errors in ce_kslope_didv based on the incoming CS waveform. 0: disabled 1: enabled
ce_ps_current_emu	7000_2400_H (ISEN) 7000_2800_H (BISEN)	[31]	Primary-side, secondary-side current sense select. 0: secondary-side current sense 1: primary-side current sense
ce_on_mask0	7000_2404_H (ISEN) 7000_2804_H	[11:0]	Defines the on-state of the current estimator with respect to the PWM outputs where the on-state generally refers to the state when the output inductor current is in its upward slope

Name	Address (loop 0/1)	Bits	Description
	(BISEN)		cycle (e.g., primary FETs on in a bridge topology). In bridge topologies there are two on-states per switching cycle. ce_on_mask0 defines the first (even) on-state and ce_on_mask1 defines the second (odd) on-state. In non-bridge topologies (e.g., ACF, buck) ce_on_mask0 and ce_on_mask1 typically are set to the same value.
ce_on_mask1	7000_2404_H (ISEN) 7000_2804_H (BISEN)	[23:12]	Defines the on-state of the current estimator with respect to the PWM outputs where the on-state generally refers to the state when the output inductor current is in its upward slope cycle (e.g., primary FETs on in a bridge topology). In bridge topologies there are two on-states per switching cycle. ce_on_mask0 defines the first (even) on-state and ce_on_mask1 defines the second (odd) on-state. In non-bridge topologies (e.g., ACF, buck) ce_on_mask0 and ce_on_mask1 typically are set to the same value.
ce_dt_l_slope	7000_2404_H (ISEN) 7000_2804_H (BISEN)	[31:24]	Defines the slope of the output inductance depending on current. Although actual inductor variation with current is non-linear this compensation introduces a linear correction to the inductor slope. It is recommended to use the 0 A inductance as one point. The user can choose the second point for the best fit curve.
ce_off_mask0	7000_2408_H (ISEN) 7000_2808_H (BISEN)	[11:0]	Defines the off-state of the current estimator with respect to the PWM outputs where the off-state generally refers to the state when the output inductor current is in its downward slope cycle (e.g., primary FETs off, secondary FETs on in a bridge topology). In bridge topologies there are two off-states per switching cycle. ce_off_mask0 defines the first (even) off-state and ce_on_mask1 defines the second (odd) off-state. In non-bridge topologies (e.g., ACF, buck) ce_off_mask0 and ce_off_mask1 typically are set to the same value.
ce_off_mask1	7000_2408_H (ISEN) 7000_2808_H (BISEN)	[23:12]	Defines the off-state of the current estimator with respect to the PWM outputs where the off-state generally refers to the state when the output inductor current is in its downward slope cycle (e.g., primary FETs off, secondary FETs on in a bridge topology). In bridge topologies there are two off-states per switching cycle. ce_off_mask0 defines the first (even) off-state and ce_on_mask1 defines the second (odd) off-state. In non-bridge topologies (e.g., ACF, buck) ce_off_mask0 and ce_off_mask1 typically are set to the same value.
ce_ltrace	7000_2408_H (ISEN) 7000_2808_H (BISEN)	[28:24]	Defines parasitic trace inductance as seen by current sense input. Computed as follows: ce_ltrace = $L_{trace}/(R_{trace} * dt)$ where, L_{trace} = parasitic trace inductance being compensated

Name	Address (loop 0/1)	Bits	Description

			R_{trace} = current sense trace resistance
			dt = 10 ns
			Example:
			L_{trace} = 300 pH, R_{trace} = 0.5 mΩ
			ce_ltrace = 300 pH/(1.0 mΩ *10 ns) = 30
			LSB = 2 H/Ω-s, range = 0 to 62 H/Ω-s

isp_scp_thresh	7000_240C_H (ISEN) 7000_280C_H (BISEN)	[7:0]	SCP fault threshold. The SCP threshold should be set highest among the various current protection thresholds as it requires only a single sample above the threshold to trip. The SCP threshold is applied per phase on multiphase topologies. Set this threshold to 0 to disable the fault detection. LSB = 1 A, range = 0 to 255 A
isp_ncl_thresh	7000_240C_H (ISEN) 7000_280C_H (BISEN)	[15:8]	Inductor negative current limit (NCL) fault threshold. The NCL threshold is compared against the instantaneous phase current. When the phase current drops below the NCL threshold the response is to turn-off the FETs associated with the off-state defined in ce_off_mask0,1. In isolated topologies this corresponds to the SR FETs. In general, turning off the SR FETs with a large negative inductor current does not produce a desirable result. It is therefore recommended to disable this feature in isolated topologies. Set to the max. positive setting, 127, to disable this fault protection and its response. LSB = 0.25 A, range = -32 to +31.75 A
ce_blank_ne_dly	7000_240C_H (ISEN) 7000_280C_H (BISEN)	[18:16]	Defines the time after a PWM pulse falling edge during which the current sense is in emulation mode only (i.e., the tracking feedback from the CS ADC is ignored). This parameter can be used to blank the ADC feedback if the falling PWM edge causes noise on the CS signal. LSB = 40 ns, range = 0 to 280 ns
ce_blank_pe_dly	7000_240C_H (ISEN) 7000_280C_H (BISEN)	[21:19]	Defines the time after a PWM pulse rising edge during which the CS is in emulation mode only (i.e., the tracking feedback from the CS ADC is ignored). This parameter can be used to blank the ADC feedback if the rising PWM edge causes noise on the CS signal. LSB = 40 ns, range = 0 to 280 ns

ce_kslope_lm	7000_2410_H
	(ISEN)
	7000_2810_H
	(BISEN)
		[8:0]	Transformer magnetizing inductance (Lm) current sense slope normalized to code/samples at 1.0 V. Only used for primary-side CS.
			ce_kslope_lm = 1.0 V * 10 ns * N_{turn}/(Lm(nH) * APC(A))
			LSB = 2^-13 V/V, range = 0.0 to 0.06238 V/V

ce_iterm	7000_2410_H (ISEN) 7000_2810_H (BISEN)	[12:9]	CS tracking error integrator coefficient. Set to 0 to disable error accumulation. LSB = 2^-13 A/A, range = 0.0 to 1.833e-3 A/A

Name	Address (loop 0/1)	Bits	Description
ce_vout_sel	7000_2410_H (ISEN) 7000_2810_H (BISEN)	[14:13]	Current sense V_{OUT} source select. Note that CDR in the table below indicates current doubler topology on the secondary.
ce_vrect_sel	7000_2410_H (ISEN) 7000_2810_H (BISEN)	[17:15]	Current sense V_{RECT} (V_{IN}) source select.

ce_current_limit	7000_2410_H (ISEN)
	7000_2810_H (BISEN)
		[26:18]	Inductor PCL fault threshold in IS ADC codes. The PWM pulse will be truncated beyond this limit. Set to 0 to disable PCL fault protection.
			ce_current_limit = current_limit(amps)/(MFR_IOUT_APC (amps/code))
			LSB = 1 code, range = 0 to 511 codes

ce_topology	7000_2410_H (ISEN) 7000_2810_H (BISEN)	[28:27]	Current emulator topology select. Defines the inductor voltage equations for V_{ON}, V_{OFF}. 0: buck, ACF, HB, FB 1: boost 2, 3: buck-boost
isp_fsw_sync_sel	7000_2410_H (ISEN) 7000_2810_H (BISEN)	[29]	Defines which loop’s frequency is used for averaging current over switching cycle. 0: loop 0 F_{switch} 1: loop 1 F_{switch} If ISEN is assigned to loop 0 and BISEN assigned to loop 1 then set isen0.isp_fsw_sync_sel=0 and isen1.isp_fsw_sync_sel=1. If both ISEN and BISEN are assigned to loop 0 then set both parameters to 0.
isp_track_fault_en	7000_2414_H (ISEN) 7000_2814_H (BISEN)	[0]	CS tracking fault enable. The tracking fault detects the inability of the CS emulator to track the incoming CS signal. This fault is typically an indication of board problem (e.g., missing sense resistor, bad pin conection). 0: disabled 1: enabled
isp_err_ratio_sel	7000_2414_H (ISEN) 7000_2814_H (BISEN)	[3:1]	CS tracking fault error ratio select. 0: 4 (11.1 percent threshold) 1: 8 (20.0 percent threshold) 2: 12 (27.3 percent threshold) 3: 16 (33.3 percent threshold) 4: 24 (42.9 percent threshold) 5: 32 (50.0 percent threshold) 6: 48 (60.0 percent threshold) 7: 64 (66.7 percent threshold)

isp_apc	7000_2418_H (ISEN)
	7000_2818_H (BISEN)
		[10:0]	CS ADC APC, maps the CS ADC LSB into amps.
			Computed by FW from PMBus command as follows:
			isp_apc = MFR_IOUT_APC
			LSB = 1.9531 mA, range = 0.0 to 3.998 A

Name	Address (loop 0/1)	Bits	Description

isp_ioffset	7000_2418$_H$
	(ISEN)
	7000_2818$_H$
	(BISEN)
		[7:0]	Phase current offset
			Computed by FW from PMBus command as follows:
			isp_offset = IOUT_CAL_OFFSET
			LSB = 0.125 A, range = -16 to +15.875 A

Common peripheral			
ical_en	7000_3000$_H$	[25]	CS sensor offset auto calibration enable. 0 = auto calibration disabled, ATE offset trim used **1 = auto calibration enabled (recommended setting)**
Telem peripheral			
tlm_in_est_alpha	7000_3400$_H$ (tlm0) 7000_3800$_H$ (tlm1)	[11:6]	Input current estimate alpha coefficient. Defines the relative contributions of V_{OUT}/V_{IN} and duty-cycle in the computation of the input current estimate.
tlm_in_scr_sel	7000_3400$_H$ (tlm0) 7000_3800$_H$ (tlm1)	[13:12]	Input current telemetry source select. 0: ISEN 1: BISEN 2, 3: estimated input current
tlm_kfp_iin	7000_3400$_H$ (tlm0) 7000_3800$_H$ (tlm1)	[19:14]	Input current telemetry LPF coefficient index. Note that exp. settings greater than 9 are clamped to 9. Set to 63 to bypass filter. • kfp_exp = tlm_kfp_iin [5:2] • kfp_man = 4 + tlm_kfp_iin [1:0] • kfp = kfp_man * 2^(kfp_exp - 13) • F3db (kHz) = [kfp/(1-kfp)] * F_{switch} (kHz)/2π
tlm_iout_scr_sel	7000_3400$_H$ (tlm0) 7000_3800$_H$ (tlm1)	[21:20]	Output current telemetry source select. 0: ISEN 1: BISEN 2, 3: sum of ISEN and BISEN (for dual-phase application)
tlm_kfp_iout	7000_3400$_H$ (tlm0) 7000_3800$_H$ (tlm1)	[27:22]	Output current telemetry LPF coefficient index. Note that exp. settings greater than 9 are clamped to 9. Set to 63 to bypass filter. • kfp_exp = tlm_kfp_iout [5:2] • kfp_man = 4 + tlm_kfp_iout [1:0] • kfp = kfp_man * 2^(kfp_exp – 13) F3db (kHz) = [kfp/(1-kfp)] * F_{switch} (kHz)/2π

3.4 PMBus command descriptions

Table 24 below describes the PMBus commands relevant to the current sense function.

XDPP1100 FW implementation:

- Page 0 PMBus commands apply to ISEN/IREF input (isen0)
- Page 1 PMBus commands apply to BISEN/BIREF input (isen1)

Table 24 CS relevant PMBus command descriptions

Command name	Command code	Format	Description
IOUT_CAL_OFFSET	39_H	LINEAR11	The IOUT_CAL_OFFSET command can be used to add a positive or negative value to the output of the output current sensing circuit to calibrate and null out any offsets. The units of the IOUT_CAL_OFFSET are amperes. This command has two data bytes formatted in the 11-bit linear data format.
MFR_IOUT_APC	EA_H	LINEAR11	Set the IOUT APC gain. The format is linear 11, -9 (exponent is -9) or linear 11, -8 (exponent is -8). The value is calculated by: MFR_IOUT_APC = ISEN_LSB/R_{sns} The ISEN_LSB is the resolution of IADC, which is determined by the isenX_gain_mode register.
FW_CONFIG_REGULATION	$C5_H$	Bit [1]	EN_PRIM_ISENSE: 0 = only ISEN enabled in single-loop design 1 = both ISEN and BISEN enabled in single-loop design

3.5 Current sense design guidelines

3.5.1 Design guidelines

For an application that has soft-switching, such as PSFB ZVS operation or ACF, it is possible to use the high-gain mode current sense. With the 100 µV resolution, the XDPP1100 can directly sense the voltage drop on a current shunt without using an external amplifier. This saves system cost and PCB layout area.

To use the high-gain mode, the XDPP1100 must be placed right next to the current sense resistor. The distance from the sense resistor to the current ADC input ISEN/IREF or BISEN/BIREF must be as short as possible. This is because the signal level is very small, only a few mV, and any switching noise or ground noise could mess up the signal. If it is not possible to place the XDPP1100 very close to the sense resistor, it is suggested to use the low-gain mode with external op-amp, which could help to increase the signal/noise level.

For a hard-switching application, it is also recommended to take the low-gain mode and use an external op-amp. The op-amp should be placed right next to the CS resistor to reduce noise pick-up.

When using PCB trace as a current shunt resistor, temperature compensation should be considered to achieve telemetry accuracy across the full temperature range. Designers should also remember to leave sufficient margin to the input signal when designing the shunt resistor and the gain of the external op-amp. For example, if the power supply is going to work in a hot environment and the PCB temperature might reach 125°C, the PCB resistance will increase by 40 percent compared to 25°C. The designer should leave 40 percent margin when calculating the maximum input signal at 25°C.

The XDPP1100 controllers can also be configured to operate with integrated power stage (IPS), which incorporates integrated current sense features (for example, Infineon's **IR3555A**). When the current sense is set to IPS mode, a 1.2 V common mode voltage should be applied to the current reference pin. The resolution of IPS mode is 1.45 mV.

3.5.2 Layout guidelines

PCB trace is often used as the CS shunt. Use the following equation to calculate copper trace resistance:

$$R_{copper} = \rho \cdot \frac{L}{T \cdot W} \cdot (1 + tc \cdot (temp - 25))$$

ρ: resistivity of copper, $17 \cdot 10^{-6}\ \Omega\ mm$

L: length of the copper trace

W: width of the copper trace

T: thickness of the copper trace

tc: temperature coefficient, $3.9 \cdot 10^{-3}$ /°C

Temp: trace temperature, unit °C

The thickness of copper trace is usually rated in ounces and represents the thickness of 1 oz. copper rolled out to an area of 1 square foot. 1 oz. copper has a thickness of 1.4 mils or 0.0356 mm. Here is a design example of the copper shunt: 130 mil x 100 mil (L x W), top layer, 4 oz. copper, trace resistance is 0.158 mΩ.

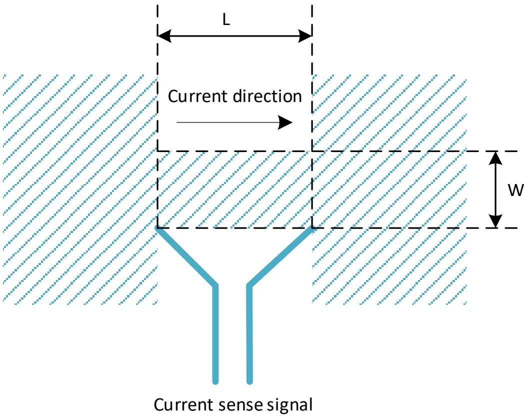


Figure 55 Current sensing by copper trace

It is recommended to lay out the copper trace shunt in the first mid-layer, so that the XDPP1100 controller can be placed right on top of the shunt trace for the shortest routing distance. Also, put a ground shielding layer next to the copper sense layer to reduce stray inductance to achieve high current sense accuracy.

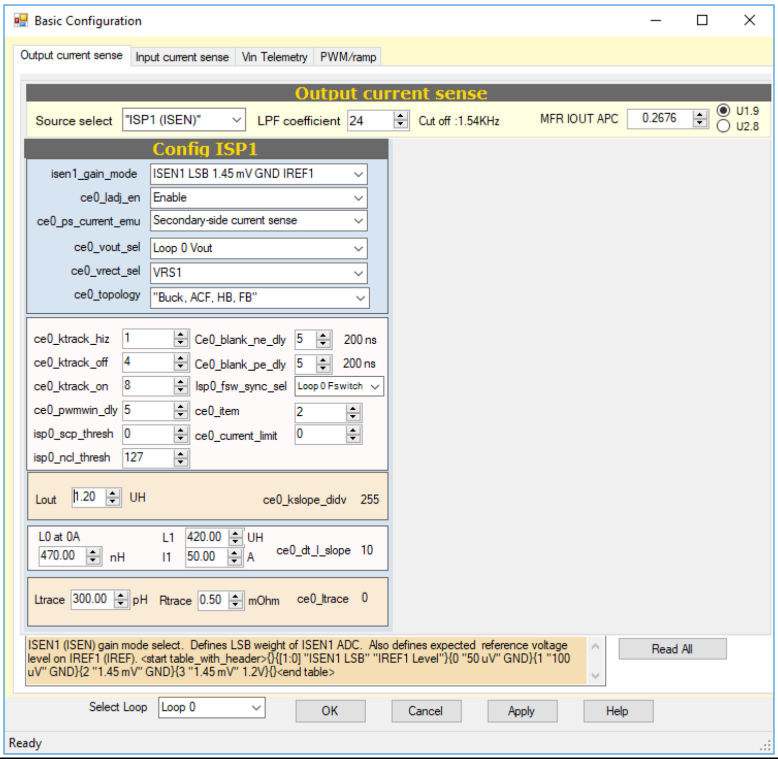
Avoid putting the current sense resistor or copper shunt next to any switching node. In high-gain mode, put the XDPP1100 as close as possible to the sense resistor. One example of good practice is putting the XDPP1100 on top of the sense resistor on the other side of the PCB. If copper shunt is used for current sense, put the temperature-sense NTC or sense diode close to the copper shunt for accurate temperature compensation.

If low-gain mode is selected, put the current sense amplifier as close as possible to the shunt resistor.

3.6 XDPP1100 GUI design tool for current sense

3.6.1 Output current sense configuration

Figure 56 shows the XDPP1100 design tool for output current telemetry.



Source select "ISP1 (ISEN)"
LPF coefficient 24
Cut off :1.54KHz
MFR IOUT APC 0.2676
● U1.9
○ U2.8

isen1_gain_mode ISEN1 LSB 1.45 mV GND IREF1
ce0_ladj_en Enable
ce0_ps_current_emu Secondary-side current sense
ce0_vout_sel Loop 0 Vout
ce0_vrect_sel VRS1
ce0_topology "Buck, ACF, HB, FB"

ce0_ktrack_hiz 1
ce0_ktrack_off 4
ce0_ktrack_on 8
ce0_pwmwin_dly 5
isp0_scp_thresh 0
isp0_ncl_thresh 127
Ce0_blank_ne_dly 5 200 ns
Ce0_blank_pe_dly 5 200 ns
Isp0_fsw_sync_sel Loop 0 Fswitch
ce0_item 2
ce0_current_limit 0

Lout 1.20 UH
ce0_kslope_didv 255

L0 at 0A 470.00 nH
L1 420.00 UH
I1 50.00 A
ce0_dt_l_slope 10

Ltrace 300.00 pH
Rtrace 0.50 mOhm
ce0_ltrace 0

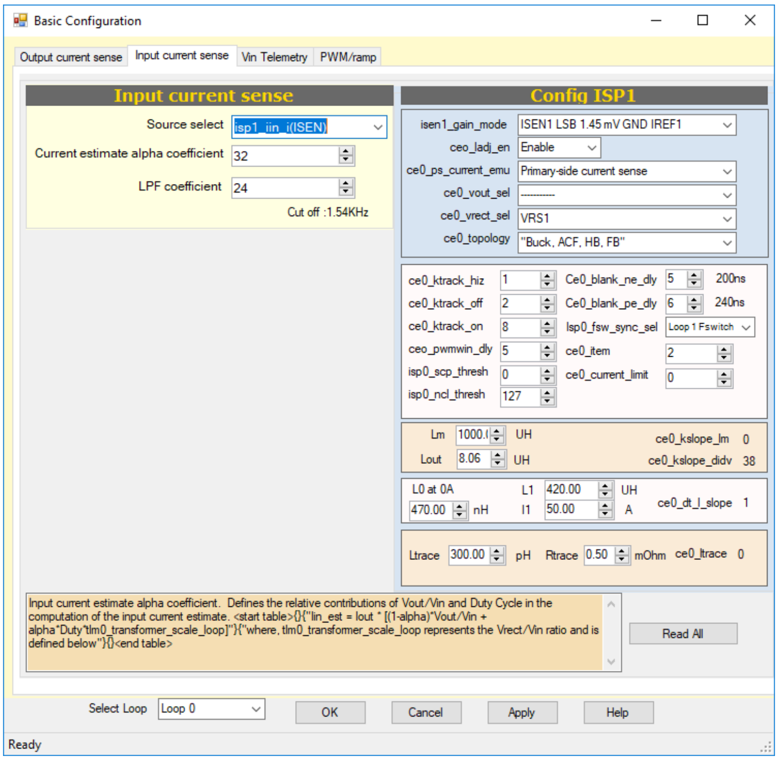
ISEN1 (ISEN) gain mode select. Defines LSB weight of ISEN1 ADC. Also defines expected reference voltage level on IREF1 (IREF). <start table_with_header>{}{[1:0] "ISEN1 LSB" "IREF1 Level"}{0 "50 uV" GND}{1 "100 uV" GND}{2 "1.45 mV" GND}{3 "1.45 mV" 1.2V}{}<end table>

Select Loop Loop 0

Figure 56 GUI design tools – output current sense

3.6.2 Input current sense configuration

Figure 57 shows the XDPP1100 design tool for input current telemetry.



Source select: isp1_iin_i(ISEN)
Current estimate alpha coefficient: 32
LPF coefficient: 24
Cut off :1.54KHz

isen1_gain_mode: ISEN1 LSB 1.45 mV GND IREF1
ceo_ladj_en: Enable
ce0_ps_current_emu: Primary-side current sense
ce0_vout_sel: ----------
ce0_vrect_sel: VRS1
ce0_topology: "Buck, ACF, HB, FB"

ce0_ktrack_hiz: 1
ce0_ktrack_off: 2
ce0_ktrack_on: 8
ceo_pwmwin_dly: 5
isp0_scp_thresh: 0
isp0_ncl_thresh: 127
Ce0_blank_ne_dly: 5 200ns
Ce0_blank_pe_dly: 6 240ns
Isp0_fsw_sync_sel: Loop 1 Fswitch
ce0_item: 2
ce0_current_limit: 0

Lm 1000.(UH
Lout 8.06 UH
ce0_kslope_lm 0
ce0_kslope_didv 38

L0 at 0A 470.00 nH
L1 420.00 UH
I1 50.00 A
ce0_dt_l_slope 1

Ltrace 300.00 pH
Rtrace 0.50 mOhm
ce0_ltrace 0

Input current estimate alpha coefficient. Defines the relative contributions of Vout/Vin and Duty Cycle in the computation of the input current estimate. <start table>{}{"Iin_est = Iout * [(1-alpha)*Vout/Vin + alpha*Duty*tlm0_transformer_scale_loop]"}{"where, tlm0_transformer_scale_loop represents the Vrect/Vin ratio and is defined below"}{}<end table>

Select Loop: Loop 0

Figure 57 GUI design tools – input current sense

4 Loop control and compensation

This chapter discusses the digital pulse width modulator and loop compensation.

4.1 PWM ramp generator

The XDPP1100 has two ramp generators and up to 12 PWM output pins. The ramp generators produce timing signals for pulse generators to create coarse resolution pulse outputs. These coarse pulses are fed to the interpolators where fine resolution timing information is used to create output pulses with 78.125 ps resolution.

The PWM module consists of two ramp generators, which support the operation of a single-loop system with up to two phases or a dual-loop system with a single phase per loop. The ramp generator produces timing information for the pulse generators for further processing. Each ramp generator receives as its input the compensation filter output. The filter used by each ramp is selected via register **rampX_pid_sel**, where X = 0 denotes ramp generator 0, and X = 1 denotes ramp generator 1. The compensation filter receives its error input from the voltage sense pins. The corresponding PID source settings are:

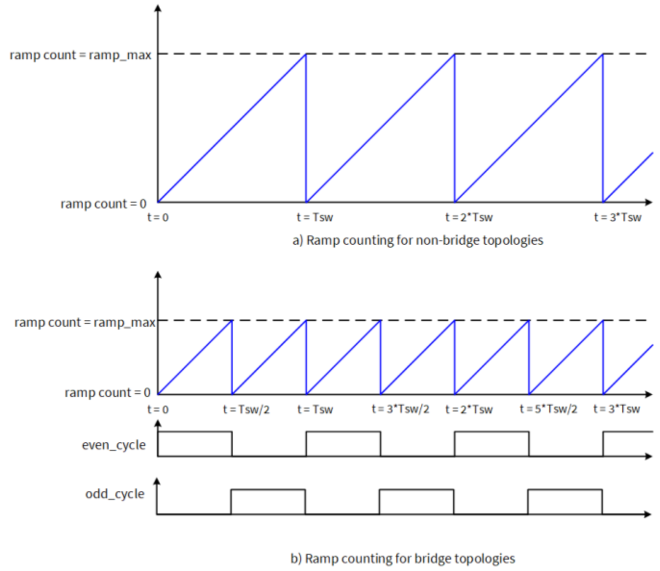
- **rampX_pid_sel** = 0 selects PID0 (VSEN)
- **rampX_pid_sel** = 1 selects PID1 (BVSEN)

Typical settings of the register **rampX_pid_sel** for the supported system configurations are shown in Table 25.

Table 25 Typical settings of the register rampX_pid_sel.

Topology	ramp0_pid_sel	ramp1_pid_sel
Single loop, single phase	0	0
Single loop, interleaved	0	0
Dual-loop	0	1

In order to generate the PWM pulses, the ramp generator produces the timing information based on a timing ramp. This timing ramp consists of a digital counter, which counts from t = 0 up to the maximum ramp count, ramp_max, before returning to 0 and counting up again. The ramp counter functionality is illustrated in **Figure 58**.



a) Ramp counting for non-bridge topologies

b) Ramp counting for bridge topologies

Figure 58 Ramp counter for a) non-bridge and b) bridge topologies

The ramp counter operation depends on the topology:

- For non-bridge topologies (buck, ACF, etc.) the ramp period is equal to the switching period (T_{switch}) and ramp_max is equal to a digital representation of T_{switch}, as shown in the upper part of **Figure 58**.
- For bridge topologies (HB, FB) the ramp period equals half of the switching period ($T_{switch}/2$) and ramp_max is equal to a digital representation of $T_{switch}/2$, as shown in the lower part of **Figure 58**.

In bridge mode, the two halves of the switching cycle are identified as either the even half-cycle or the odd half-cycle, as shown at the bottom part of **Figure 58**. This identification is used by the pulse generator to produce pulses only on the correct half-cycle.

Register **rampX_half_mode** (X = 0, 1) is used to select between bridge and non-bridge topologies. This register defines whether half mode is enabled for the ramp. If the mode is enabled, the ramp count (ramp_max) equals half of the switching period. Correspondingly, if half mode is disabled, the ramp count equals the switching period.

The switching period, T_{switch}, is defined by the register **tswitchX** (X = 0, 1). It is automatically programmed by the FW based on the PMBus command FREQUENCY_SWITCH, which sets the switching frequency in kHz. The register **tswitchX** has LSB weight of 20 ns and range 0.0 to 10.22 µs. Therefore, there are some considerations regarding FREQUENCY_SWITCH values:

- Only values with corresponding T_{switch} that are an exact multiple of 20 ns can be achieved
- If FREQUENCY_SWITCH is set to a value that cannot be achieved in **tswitchX**, the FW will choose the closest achievable setting

The closest achievable FREQUENCY_SWITCH to a target F_{switch} can be found according to Equation 4.1.

$$FREQUENCY\_SWITCH = \frac{50e6}{ROUND\left(\frac{50e6}{target\ F_{switch}}\right)}, \text{Note: } 50e6 = 1/20e\text{-}9 \qquad (4.1)$$

4.1.1 PWM ramp modulation schemes

The timing information provided by the ramp generator is a pair of timing markers called t1 and t2. These timing markers are used by the pulse generator to define the rising and falling edges of the PWM pulses.

The XDP1100 supports the following modulation schemes:

- Trailing edge (TE) modulation
- Leading edge (LE) modulation
- Dual edge (DE) modulation

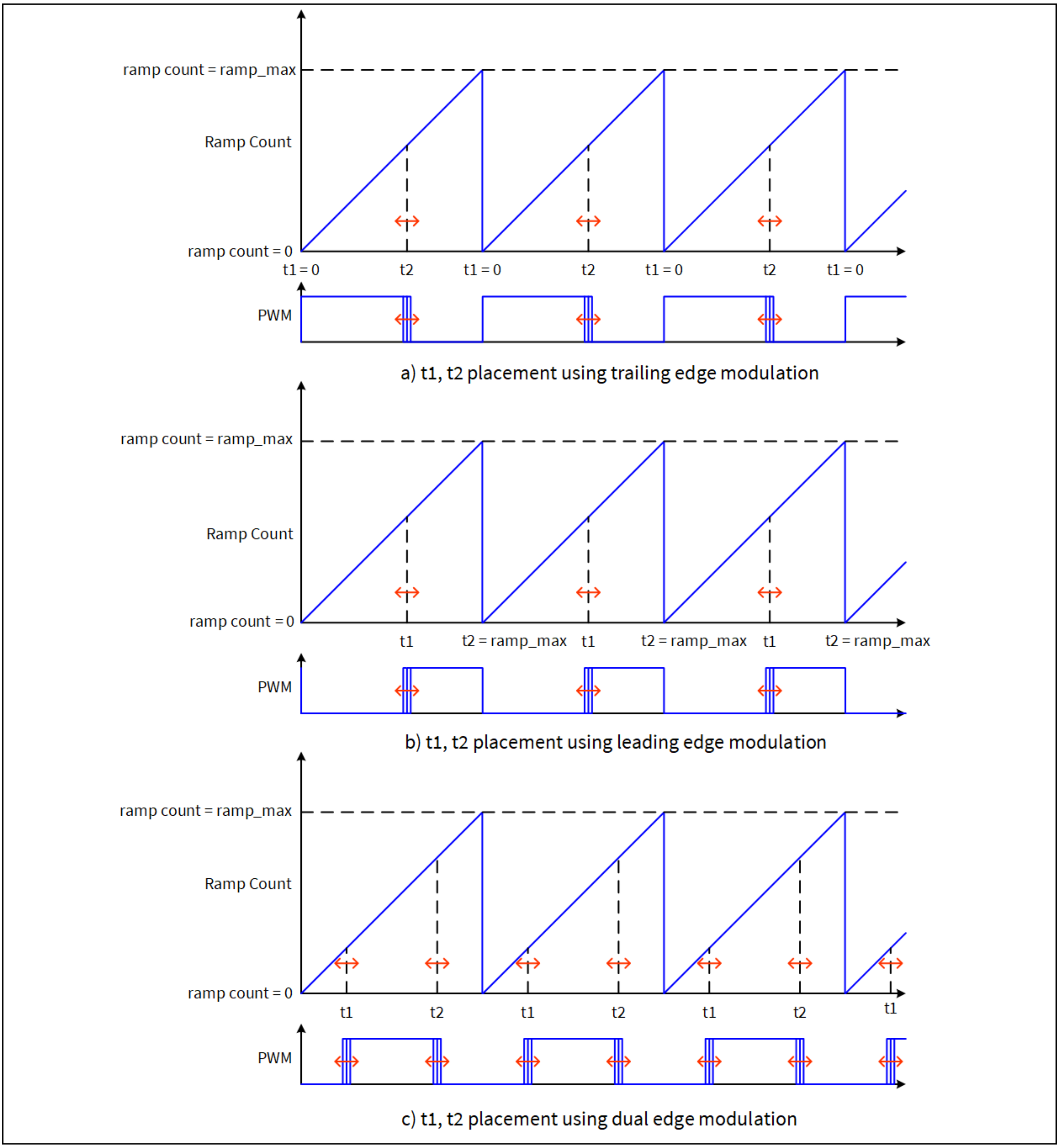
These modulation schemes are shown in **Figure 59**. The first modulation waveform in **Figure 59** a) shows the trailing edge modulation case. The timing marker placement for TE modulation:

- t1 is fixed at ramp count = 0
- t2 is modulated based on the selected feedback control mode

The second modulation scheme is LE modulation, shown in **Figure 59** b). The timing marker placement for LE modulation is the following:

- t1 is modulated based on the selected feedback control mode
- t2 is fixed at ramp count = ramp_max

The last modulation scheme is DE modulation, shown in **Figure 59** c). In this modulation, the timing markers t1 and t2 are both modulated based on the selected feedback control mode. The example PWM pulse shown in **Figure 59** c) has modulated leading and trailing edge.



a) t1, t2 placement using trailing edge modulation

b) t1, t2 placement using leading edge modulation

c) t1, t2 placement using dual edge modulation

Figure 59 **T1 and t2 placement for TE, LE and DE modulation**

The pulse width, t2 to t1, for all modulation schemes is given in Equation 4.2:

$$t_2 - t_1 = D * ramp\_max \tag{4.2}$$

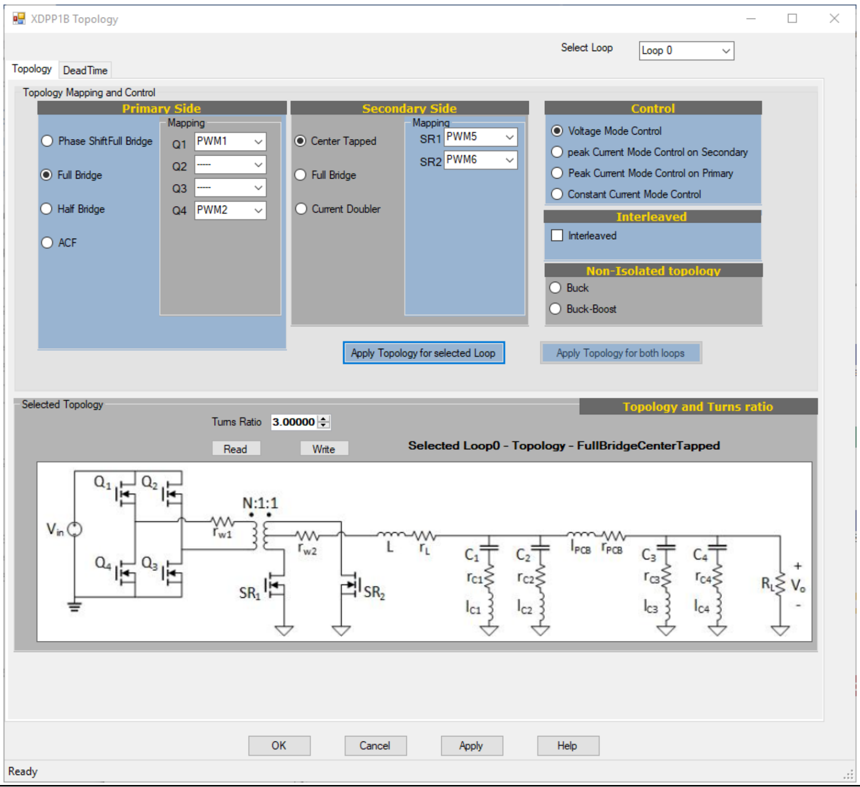
D is the duty-cycle output of the compensation filter in the case of VMC. It should be noted that the pulse generator is also capable of creating a pulse with leading edge at t2 and trailing edge at t1. This pulse would have a pulse width equal to (1 – D) * ramp_max. The modulation scheme for both ramps is selected via register **rampX_m_flavor** (X = 0,1) as shown in Table 26.

Table 26 Modulation type settings

rampX_m_flavor	Modulation type
0	Dual edge (DE)
1	Leading edge (LE)
2, 3	Trailing edge (TE)

4.1.2 PWM mapping

The XDPP1100 GUI has a design tool to help the user select a topology and assign PWM to drive primary and secondary MOSFETs. **Figure 60** shows the topology and PWM mapping design tool.



XDPP1B Topology

Select Loop: Loop 0

Topology Mapping and Control

Primary Side
○ Phase ShiftFull Bridge
● Full Bridge
○ Half Bridge
○ ACF

Mapping
Q1: PWM1
Q2: -----
Q3: -----
Q4: PWM2

Secondary Side
● Center Tapped
○ Full Bridge
○ Current Doubler

Mapping
SR1: PWM5
SR2: PWM6

Control
● Voltage Mode Control
○ peak Current Mode Control on Secondary
○ Peak Current Mode Control on Primary
○ Constant Current Mode Control

Interleaved
□ Interleaved

Non-Isolated topology
○ Buck
○ Buck-Boost

Selected Topology

Topology and Turns ratio

Turns Ratio: 3.00000

Selected Loop0 - Topology - FullBridgeCenterTapped

Ready

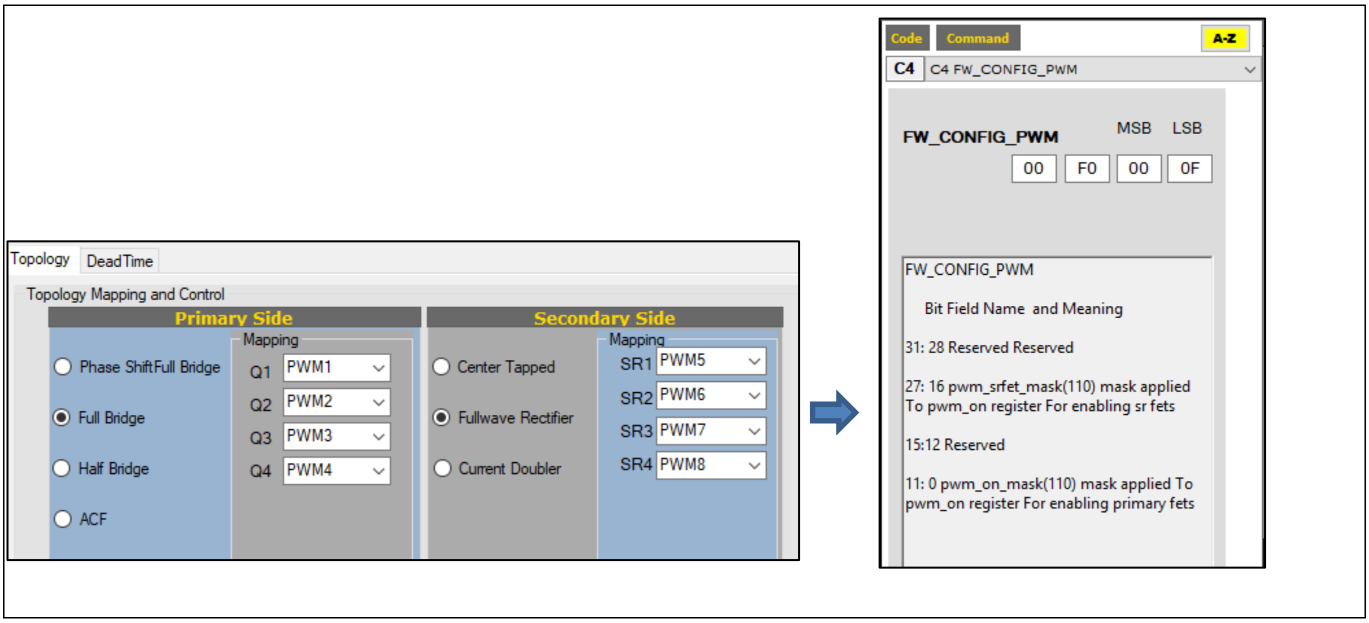
Figure 60 GUI design tool – topology and PWM mapping

The user can select isolated or non-isolated topology from the listed options. Once the topology is selected, the tool will show the schematic of the topology at the bottom section of the window. The primary MOSFETs are labeled as Qx and the secondary SR MOSFETs are labeled as SRx.

Use the mapping tool to assign a PWMx signal to each MOSFET. The XDPP1100 enables mapping all the PWMs freely without restrictions to primary or secondary. For FB configuration, the diagonal MOSFETs can be driven by the same PWM signal. For example, Q1 and Q3 in **Figure 60** are a pair of diagonal MOSFETs. The topology tool allows leaving the Qx or SRx un-mapped. **Figure 60** is an example showing that Q2 and Q3 are not assigned to any PWM output. The user should take care of the driver circuit with HW when sharing the same PWM for the diagonal switches.

Once the topology and PWM are configured and applied to the design, the GUI assigns the t1 and t2 crossing of selected PWM per the topology and the location of the MOSFET. The GUI also write the PMBus command FW_CONFIG_PWM based on the PWM mapping. The FW_CONFIG_PWM defines the PWMx mask of primary FETs and secondary SR FETs.

Taking the FB-FB example, the PWMs are mapped per **Figure 61**. PWM1-PWM4 are mapped to primary FETs, thus the last two bytes of FW_CONFIG_PWM are 00 0F. PWM5-PWM8 are mapped to SR FETs, thus the first two bytes are 00 F0. It is not recommended to manually change the PWM mask with this command unless non-standard topology is used, and the user wants to manully configure the PWM mask.



Topology Mapping and Control

Primary Side

○ Phase ShiftFull Bridge
● Full Bridge
○ Half Bridge
○ ACF

Mapping
Q1 PWM1
Q2 PWM2
Q3 PWM3
Q4 PWM4

Secondary Side

○ Center Tapped
● Fullwave Rectifier
○ Current Doubler

Mapping
SR1 PWM5
SR2 PWM6
SR3 PWM7
SR4 PWM8

C4 C4 FW_CONFIG_PWM

FW_CONFIG_PWM MSB LSB
00 F0 00 0F

FW_CONFIG_PWM

Bit Field Name and Meaning

31: 28 Reserved Reserved

27: 16 pwm_srfet_mask(110) mask applied To pwm_on register For enabling sr fets

15:12 Reserved

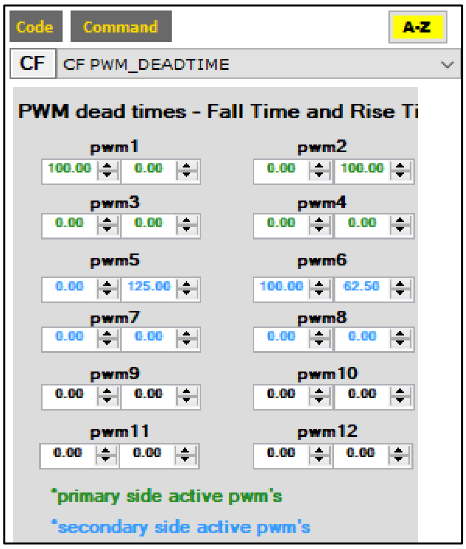
11: 0 pwm_on_mask(110) mask applied To pwm_on register For enabling primary fets

Figure 61 FW_CONFIG_PWM

4.1.3 Dead-time configuration

PWM dead-time can be set by PMBus command 0xCF PWM_DEADTIME (**Figure 62**). The dead-times of PWM rise and fall time can be set separately. The rise time of dead-time adds dely to the rising edge of PWM. The fall time of dead-time adds dely to the falling edge of PWM. The dead-time is always positive. As PWM_DEADTIME defines the dead-time of all 12 PWMx, it is a common command and applies to both loops. In the XDPP1100 GUI, the active primary and secondary PWMs will be highlighted in green and blue respectively, as shown in **Figure 62**. Writing inactive PWMs is not allowed in the GUI to prevent accidental setting of the dead-time of the other loop.

The maximum dead-time can be set to 318.75 ns with a resolution of 1.25 ns.



CF | CF PWM_DEADTIME

PWM dead times - Fall Time and Rise Ti

pwm1: 100.00 | 0.00
pwm2: 0.00 | 100.00
pwm3: 0.00 | 0.00
pwm4: 0.00 | 0.00
pwm5: 0.00 | 125.00
pwm6: 100.00 | 62.50
pwm7: 0.00 | 0.00
pwm8: 0.00 | 0.00
pwm9: 0.00 | 0.00
pwm10: 0.00 | 0.00
pwm11: 0.00 | 0.00
pwm12: 0.00 | 0.00

*primary side active pwm's
*secondary side active pwm's

Figure 62 PWM_DEADTIME

When setting the dead-time of isolated topologies, please consider the isolator delay, which will be added to the actual primary gate driver. Setting the rise time delay of the SR PWM longer than the isolator delay is recommended.

4.2 Feedback control modes

The XDPP1100 supports VMC and PCMC. In the case of PCMC, both primary-side and secondary-side control are possible. The feedback control mode is selected via register **mode_control_loopX**, where X denotes zero or one depending on which loop is being used. The control method of programming is shown in Table 27.

Table 27 Feedback control mode programming

mode_control_loopX	Feedback control mode
0	Voltage mode control (VMC)
1	Peak current mode control (PCMC) on secondary
2	Peak current mode control (PCMC) on primary
3	Reserved

4.2.1 VMC

VMC is the simplest control method, and it can be configured by setting the register value **mode_control_loopX** to 0, as described in Table 27. The functional block diagram shown in **Figure 63** illustrates the internal configuration while the XDPP1100 applies VMC.

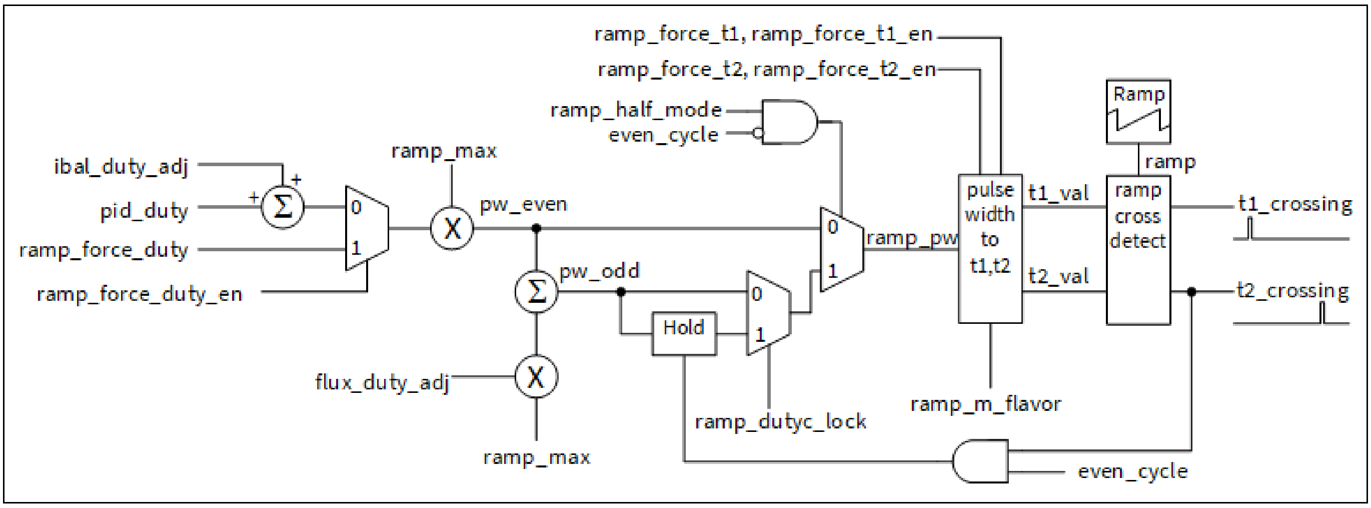


Figure 63 Functional block diagram of VMC

In VMC the compensator output, pid_duty, is directly interpreted as the duty-cycle with a range of 0.0 to 0.9999. In addition, two other sources contribute to the final duty-cycle used to compute the PWM pulse width (PW). These sources are:

- Current balance duty-cycle adjustment (ibal_duty_adj). This signal comes from the current balance function (described in Chapter 8). It corrects for difference in the phase current in a dual-phase (interleave) loop and it is applied only to ramp 0. The phase associated with ISEN contains the current balance duty-cycle adjustment while the phase associated with BISEN does not.
- The flux balance duty-cycle adjustment (fbal_duty_adj). This signal comes from the flux balance function (described in Chapter 6). It corrects for the transformer flux (volt-second) differences between the even and odd half-cycles in the FB topology and it is applied only to the odd half-cycle.

As shown in **Figure 63**, the duty-cycle components are multiplied by ramp_max to convert from duty-cycle to PW, ramp_pw. The PW is then converted to target t1 and t2 values, t1_val and t2_val, based on the modulation type, as shown in Table 28.

Table 28 T1 and t2 computation by edge modulation type

Modulation type	t1_val	t2_val
Dual edge (DE)	(ramp_max - ramp_pw)/2	(ramp_max + ramp_pw)/2
Leading edge (LE)	ramp_max - ramp_pw	ramp_max
Trailing edge (TE)	0	ramp_pw

The target t1 and t2 values are then compared against the ramp waveform to create the t1_crossing and t2_crossing signals used by the pulse generators to define the PWM edges.

Duty-cycle lock mode is intended for use in FB topologies where it is important to maintain the flux balance between the two half-cycles. In this mode, the odd half-cycle PW is sampled and held on the even half-cycle t2 crossing detection. The duty-cycle lock mode is enabled by setting the register **rampX_dutyc_lock** to 1.

It should be noted that the flux balance adjustment still contributes to the odd cycle PW even when duty-cycle lock is enabled. This means that the PWs will not be identical if compensation is required to correct some external deviation (e.g., differences in driver propagation delays).

4.2.2 PCMC

PCMC is based on current information and it can be selected by setting the register value **mode_control_loopX** to 1 for secondary-side PCMC or to 2 for primary-side PCMC.

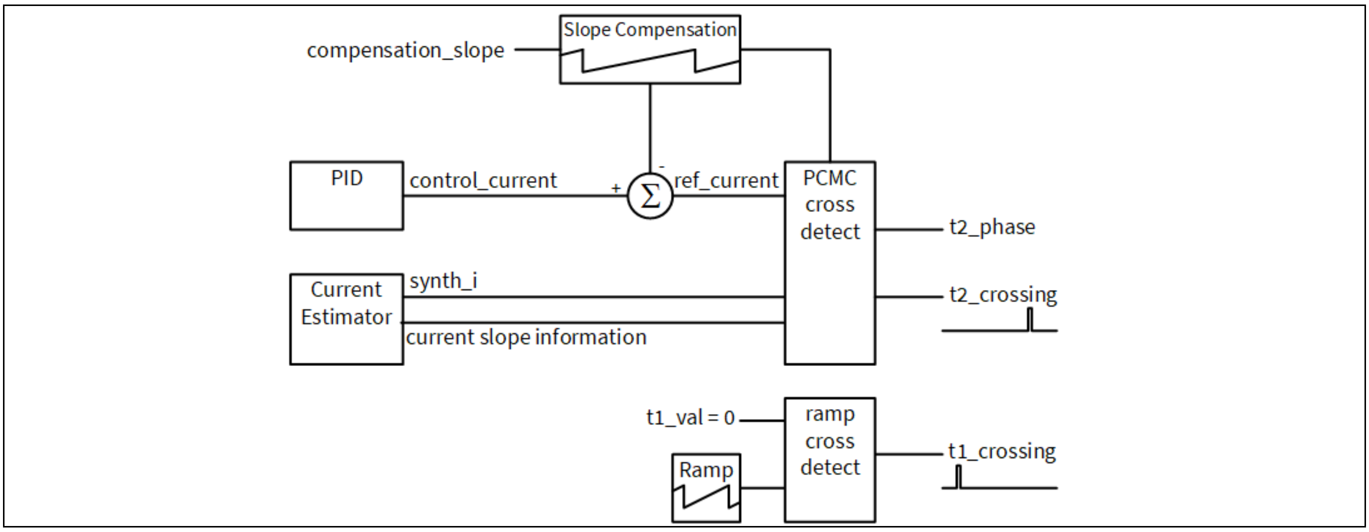


Figure 64 Functional block diagram of PCMC

In the case of PCMC the compensator output represents the control current, whereas for VMC it was directly the duty-cycle. **Figure 64** shows the functional block diagram of the XDPP1100 internal configuration when PCMC is applied.

PID selection for ramp0 and ramp1 is programmable, whereas the current estimator selection is hardwired. This references that:

- Current estimator CE0 (ISEN) is connected to ramp 0
- Current estimator CE1 (BISEN) is connected to ramp 1
- Current used for control (either primary or secondary) should be sensed by ISEN/IREF in a single-loop system using ramp 0

The PID output represents “normalized control current”, where the normalization is to the maximum range of the IS ADC. This leads to a signed output range of -1.0 to +1.0. PCMC supports only TE modulation, and therefore the register **rampX_m_flavor** should be set to value 2. This means that the t1 crossing event always occurs at ramp = 0.

PCMC requires slope compensation to prevent sub-harmonic oscillation when the duty cyle is higher than 0.5. A slope compensation ramp is provided, and its value is programmable through register **compensation_slope**, as shown in Table 29.

Table 29 PCMC slope compensation values

Compensation_slope	Slope of compensation ramp
0	V_{OUT}/L
1	$V_{OUT}/2L$
2	$V_{OUT}/4L$
3	Reserved

As illustrated in **Figure 64**, the output of the slope compensation ramp is subtracted from the control current in order to obtain the reference current. The reconstructed current output from the current estimator, synth_i, is then compared against this reference current. Both of these currents are digital and discrete in time. Therefore, in order to determine the convergence rate of these two signals, the PCMC cross-detect function uses the slope information from both the slope compensation ramp and the current estimator. This convergence rate information is used to:

- Predict the t2 crossing in the next clock cycle
- Determine the phase of the t2 crossing within the clock cycle

The phase is used by the interpolators for fine timing resolution. It should be noted that in the PCMC the PWM pulse width resolution is limited to 625 ps compared to the 78.125 ps available in VMC.

4.2.3 Maximum and minimum PW enforcement

The maximum pulse width (PW) is possible to enforce. This can be performed either by using a fixed or variable method. These methods are adjusted in the following ways:

- Fixed method: Register **rampX_dc_max** defines the fixed maximum duty-cycle for ramp X, where the register value is computed by FW from PMBus command MAX_DUTY, as shown in Equation 4.3.
- Variable method: Register **rampX_dc_max_nom** defines the variable maximum duty-cycle limit for ramp X, where the variable limit scales the maximum duty-cycle with the measured V_{RECT} in order to limit the transformer flux at high V_{IN}. The scaling is with respect to register **pid_vrect_ref**, as given in Equation 4.4.

$$fixed\_\max\_duty = rampX\_dc\_max = MAX\_DUTY \tag{4.3}$$

$$variable\_max\_duty = rampX\_dc\_\max\_nom * \frac{pid\_vrect\_ref}{Vrect} \tag{4.4}$$

Setting **rampX_dc_max_nom** to 0 disables the variable maximum duty-cycle limit. The applied maximum duty limit is the minimum of the fixed and variable limits assuming the variable limit is enabled, as shown below.

```
if (rampX_dc_max_nom>0)
       max_duty = MIN(fixed_max_duty, variable_max_duty)
else
       max_duty = fixed_max_duty
```

The maximum PW limit is computed as the product of the maximum duty-cycle limit and the maximum ramp value according to Equation 4.5.

$$pw\_max = \max\_duty * rampX\_max \tag{4.5}$$

The minimum PW is defined by register **rampX_pw_min** and its value is automatically computed by the FW based on the PMBus command MFR_MIN_PW according to Equation 4.6.

$$rampX\_pw\_min = MFR\_MIN\_PW \tag{4.6}$$

Setting this register value to 0 disables the minimum PW enforcement. Two methods of minimum PW enforcement are supported in VMC. They can be defined through register **rampX_min_pw_state** in the following way:

- Register value 0, sets PW to 0 (blank pulse)
- Register value 1, sets PW to **rampX_pw_min** value (clamp pulse to min.)

In PCMC only the clamp to min. method is supported.

4.3 Compensator

Most parts of this chapter assume that compensator output is directly the duty-cycle, as in VMC, and in Section 4.4 the PCMC is presented.

The simplified block diagram of the compensator is shown in **Figure 65**, and it consists of:

- Compensation filter, shown within the dashed lines
- Input voltage feed-forward (FF) function

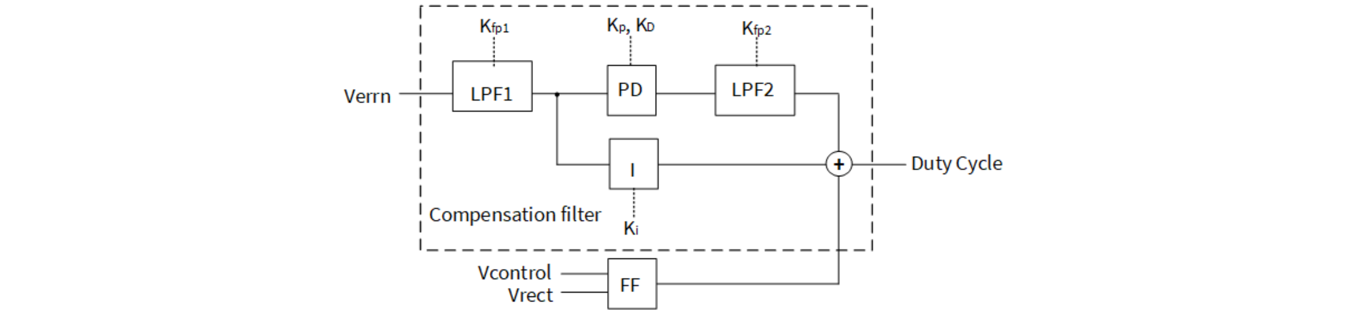


Figure 65 Functional block diagram of the compensator

The compensation filter, inside the dashed lines in **Figure 65**, consists of:

- Proportional-integral-derivative (PID) filter
- Pre- and post-low pass filters (LPF)

It receives as input the computed error signal, V_{ernn}, obtained from the voltage sense processor. The compensator output is the target duty-cycle, utilized by the PWM to generate output pulses.

The transfer function implemented by the compensation filter is provided in Equation (4.7).

$$\frac{DutyCycle}{Verrn} = \left[\frac{K_{fp1}}{1-(1-K_{fp1})z^{-1}}\right]\left[\left(K_P + K_D(1-z^{-1})\right)\left(\frac{K_{fp2}}{1-(1-K_{fp2})z^{-1}}\right) + \frac{K_i}{1-z^{-1}}\right] + FF \qquad (4.7)$$

The terms K_p, K_D and K_i are the PID loop coefficients and $K_{fp1,2}$ the LPF coefficients, which determine the locations of the poles and zeroes. The zero locations are defined by the coefficients K_p, K_i and K_D, which are programmed via the following registers:

- **pid_kp_index_1ph** for K_p
- **pid_ki_index_1ph** for K_i
- **pid_kd_index_1ph** for K_D

The PID provides a pole in the origin and two mid-band zeroes, while the LPFs present two high-frequency poles. This corresponds to the type III compensation response, which can provide up to 180 degrees of phase boost. A typical type III compensator gain is illustrated in **Figure 66**, where f_{z1}, f_{z2}, f_{p1} and f_{p2} represent the pole and zero frequencies.

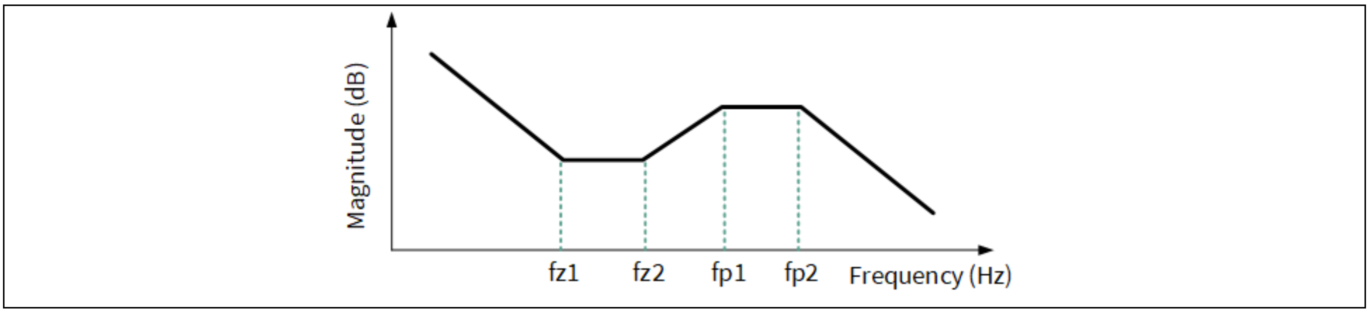


Figure 66 Typical type III compensator gain

4.3.1 Pre-filter

The pre-filter is a single pole LPF, implemented as shown in **Figure 67**, and it consists of:

- One input, the error voltage V_{ernn}
- Two outputs, verrn_filt and verrn_slope

The verrn_filt is the low-pass filtered version of the computed error, verrn, and it is used by the proportional and integral terms of the PID. The second output, verrn_slope, is the derivative of the filtered error and it is used by the derivative term of the PID.

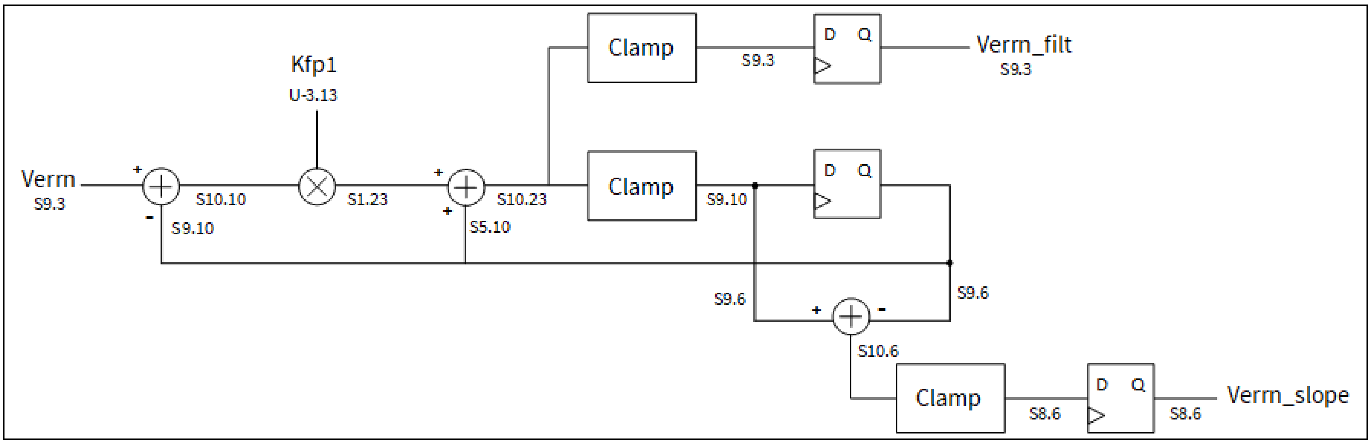


Figure 67 Pre-filter block diagram

The pre-filter creates the first high-frequency pole, f_{p1}, in the compensator transfer function. The pole location (i.e. LPF bandwidth) is defined by the filter coefficient K_{fp1}, which is programmed via register **pid_kfp1_index_1ph**.

The filter coefficient parameters use an exponent mantissa format to provide an extended range, utilizing fewer total bits. The upper three bits of the kfp1_index represent the exponent and the lower three bits represent the mantissa. The integer and real number representations of K_{fp1} are computed as shown below.

$$k_{fp1}\_exp = (\ k_{fp1}\_index[5:3]) \tag{4.8}$$

$$k_{fp1_{man}} = k_{fp1}\_index[2:0] \tag{4.9}$$

$$K_{fp1} = (8 + k_{fp1}\_man) * 2^{k_{fp1}\_exp} \tag{4.10}$$

$$K_{fp1}\_real = K_{fp1} * 2^{-13} \tag{4.11}$$

The location of the first high-frequency pole can be computed as shown in Equation 4.12, using the previously computed real number representation for K_{fp1}.

$$fp1 = \left(\frac{1}{2\pi T_s}\right)\left(\frac{K_{fp1\_real}}{1-K_{fp1\_real}}\right), T_s = \frac{1}{50MHz} \tag{4.12}$$

Note that **pid_kfp1_index_1ph** is clamped to 55 internally, corresponding to a maximum exponent of 6.

Figure 68 shows the pre-filter bandwidth as a function of the parameter kfp1_index. The post-filter uses the same index to coefficient mapping as the pre-filter so the coefficient discussion in this section also applies to the post-filter (K_{fp2} and kfp2_index).

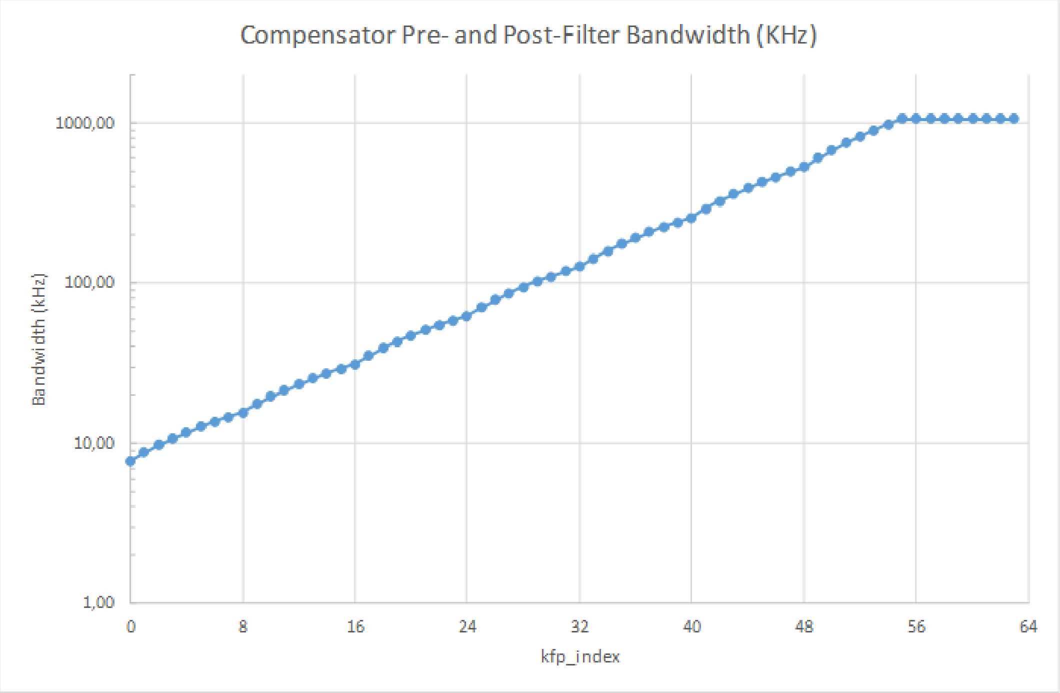


Figure 68 **Pre- and post-filter bandwidth as a function of kfp_index**

4.3.2 PID term computation

The PID coefficients are computed based on the pre-filter output signals:

- Proportional (p_term) and integral (i_term) terms are obtained from the filtered error signal verrn_filt
- Derivative term (d_term) is computed based on verrn_slope, the derivative of the filtered error signal

The p_term and d_term are added to produce the pd_term, which is downstream processed by the post-filter. The high-frequency gain of the integral term is negligible compared to the proportional and derivative terms. Therefore, no need exists for further low-pass filtering. The computation block diagram for PID terms is shown in **Figure 69**.

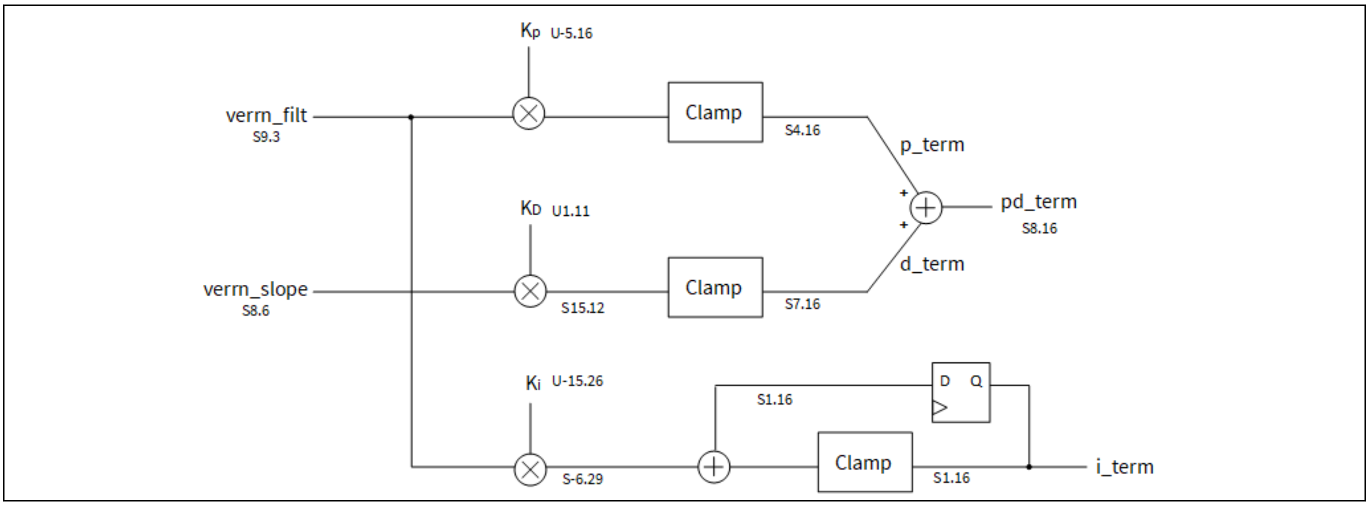


Figure 69 PID computation block diagram

Similarly to the pre-filter, the PID parameters use an exponent mantissa format to provide an extended range using fewer total bits:

- The upper three bits of kp_index and ki_index represent the exponent and the lower three bits represent the mantissa
- For kd_index, the upper four bits represent the exponent and the lower three bits represent the mantissa

The integer and real number representations of K_p are computed as shown in the following equations:

$$k_P\_exp = k_P\_index[5:3] \tag{4.13}$$

$$k_P\_man = k_P\_index[2:0] \tag{4.14}$$

$$K_P = (8 + k_P\_man) * 2^{k_P\_exp} \tag{4.15}$$

$$K_p\_real = K_P * 2^{-16} \tag{4.16}$$

Correspondingly, the integer and real number representations of K_i are computed as given in Equations 4.17 to 4.20.

$$k_i\_exp = k_i\_index[5:3] \tag{4.17}$$

$$k_i\_man = k_i\_index[2:0] \tag{4.18}$$

$$K_i = (8 + k_i\_man) * 2^{k_i\_exp} \tag{4.19}$$

$$K_i\_real = K_i * 2^{-26} \tag{4.20}$$

The integer and real number representations of K_D are computed as shown in Equations 4.21 to 4.24. Note that kd_index is clamped to 119 internally, corresponding to a maximum exponent of 14.

$$k_D\_exp = MIN(14, k_D\_index[6:3]) \tag{4.21}$$

$$k_D\_man = (k_D\_index[6:3] > 14)?\,7 : k_D\_index[2:0] \tag{4.22}$$

$$K_D = (8 + k_D\_man) * 2^{k_D\_exp} \tag{4.23}$$

$$K_D\_real = K_D * 2^{-11} \tag{4.24}$$

The PID creates two mid-band zeroes, f_{z1} and f_{z2}, in the compensator transfer function. Based on the real number representation of the PID coefficients, the location of the zeroes can be obtained according to Equations 4.25 to 4.26, where T_S equals $\frac{1}{50MHz}$.

$$f_{z1} = \left(\frac{1}{2\pi T_s}\right)\left(K_P\_real - \sqrt{K_P\_real^2 - 4K_d\_real * K_i\_real}\right)/2K_d\_real \tag{4.25}$$

$$f_{z2} = \left(\frac{1}{2\pi T_s}\right)\left(K_P\_real + \sqrt{K_P\_real^2 - 4K_d\_real * K_i\_real}\right)/2K_d\_real \tag{4.26}$$

4.3.3 Post-filter and summation

The post-filter is a single pole LPF and it receives its input, pd_term, from the PID filter. The integral term has negligible gain compared to the proportional and derivative terms at high frequency, thus no additional filtering is needed.

The compensator duty-cycle output is created by adding the post-filtered output to:

- Integral term output, i_term
- Voltage feed-forward term, ff_duty

The implementation of the post-filter is illustrated in **Figure 70**.

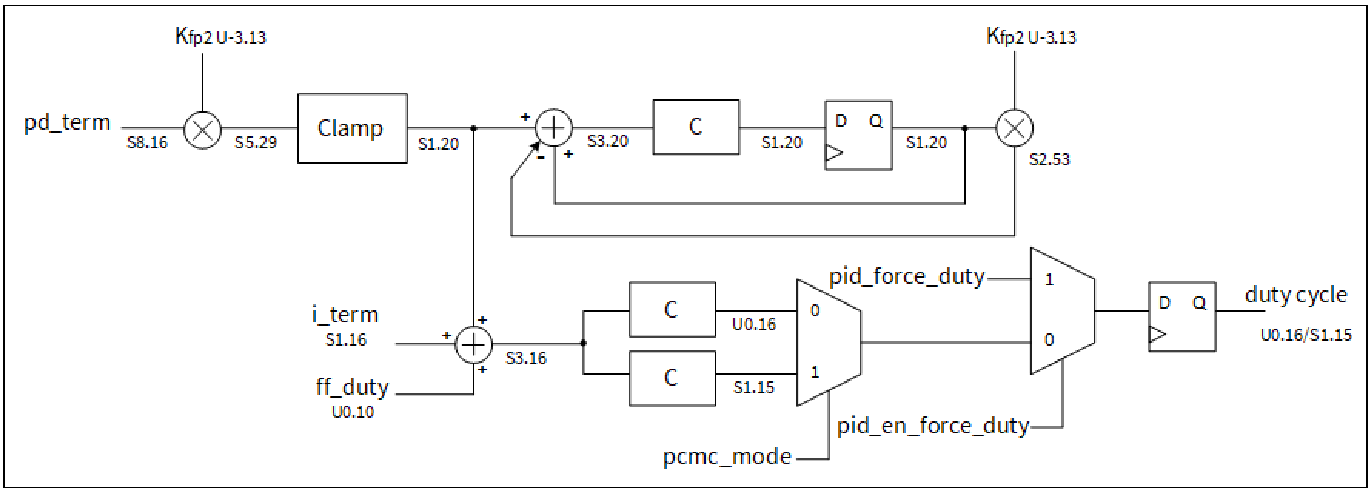


Figure 70 Post-filter block diagram

The post-filter creates the second high-frequency pole, f_{p2}, in the compensator transfer function. The pole location (i.e., LPF bandwidth) is defined by the filter coefficient K_{fp2}, which is programmable through register **pid_kfp2_index_1ph**.

Similarly to the pre-filter, kfp2_index uses exponent mantissa format to provide an extended range using fewer total bits. The corresponding integer and real number representations of K_{pf2} for each kfp2_index value are the same as for K_{pf1} provided in the pre-filter, see 4.3.1. Correspondingly, the post-filter bandwidth as a function of parameter kfp2_index is the same as shown in **Figure 68**.

4.3.4 Input/output clamping of the compensation filter

The input and the output of the PID filter can be clamped. The input clamp is user-programmable through register **vsp_verrn_clamp_thresh**, and it limits the maximum error seen by the compensation filter.

The compensation filter output, duty_cycle, is clamped to a fixed 16-bit width with. In the case of VMC it references a value between 0.0 and 1.0. The PCMC behavior and the corresponding compensator output value are discussed in Section 4.4.

4.3.5 Output override – forced duty cycle

The PID output of the compensation filter can be overridden. The user can force duty-cycle value by selecting:

- Register **pid_force_duty_en**
- Setting a value in register **pid_force_duty**

This functionality is illustrated in the post-filter block diagram in **Figure 70**.

Since this override is applied at the PID output, downstream adjustments to the duty-cycle are still applied. These adjustments include:

- Current balance in an interleaved (multiphase) design
- Flux balance in a FB design

However, if the downstream adjustments are required to be overridden, a separate pair of override registers is provided:

- **ramp0_force_duty**
- **ramp1_force_duty**

4.3.6 Coefficient scaling

The PID coefficients are scaled with V_{RECT} in order to maintain constant loop gain despite the input voltage variations. The user can define a reference V_{RECT} voltage through register **pid_vrect_ref**, at which the gain scale is 1.0. This coefficient scale factor is defined as shown in Equation 4.27:

$$Coefficient\ scale\ factor = \frac{V_{RECT}}{pid\_vrect\_ref} \tag{4.27}$$

The register **pid_vrect_ref** value should be set to the expected nominal V_{RECT} voltage prior to the PID coefficient optimization.

An example for selecting a proper **pid_vrect_ref** value is given below for a FB topology with a nominal input voltage of 48 V and transformer turns ratio of 3. The nominal rectified voltage is computed in Equation 4.28 and the resulting value should be set via register **pid_vrect_ref**. Please note that for HB topology, the voltage applied to the transformer primary winding is half of the input voltage and thus V_{RECT_nom} should be divided by 2 as in Equation 4.28. The design tool in the GUI calculates the **pid_vrect_ref** based on the topology and the transformer scale is user defined, thus the user just needs to provide the value of nominal V_{IN}.

$$V_{RECT-nom} = \frac{V_{in-nom}}{N_{turns}} = \frac{48V}{3} = 16\ V \tag{4.28}$$

Subsequently, the optimized PID coefficients can be observed from registers:

- **pid_kp_eff**, for K_p
- **pid_ki_eff**, for K_i
- **pid_kd_eff**, for K_D

4.3.7 Freeze, reset accumulator

Undesired integrator "windup" could occur in the following operating conditions:

- Burst ,ode operation, where PID is not controlling the output duty-cycle
- Duty-cycle output is 0 and a negative verrn input is received
- Peak current limit (PCL) has been exceeded and a positive verrn is received
- PW exceeds the maximum PW (duty-cycle exceeds the max. duty-cycle) and a positive verrn is received

Under these conditions, the XDPP1100 hardware freezes the integrator term accumulator.

In addition to the HW freeze conditions, it is possible to freeze the integrator accumulator through register **pid_freeze_accum**. Another register, **pid_reset_accum**, allows the FW to reset the integrator accumulator to 0.

4.4 Control mode selection – peak current mode

The difference between VMC and PCMC control methods is that for the VMC, the compensator output is directly the duty-cycle, whereas for the PCMC the compensator output is a reference current that is compared to the sensed current. Therefore, depending on the desired control method, the compensator output format varies:

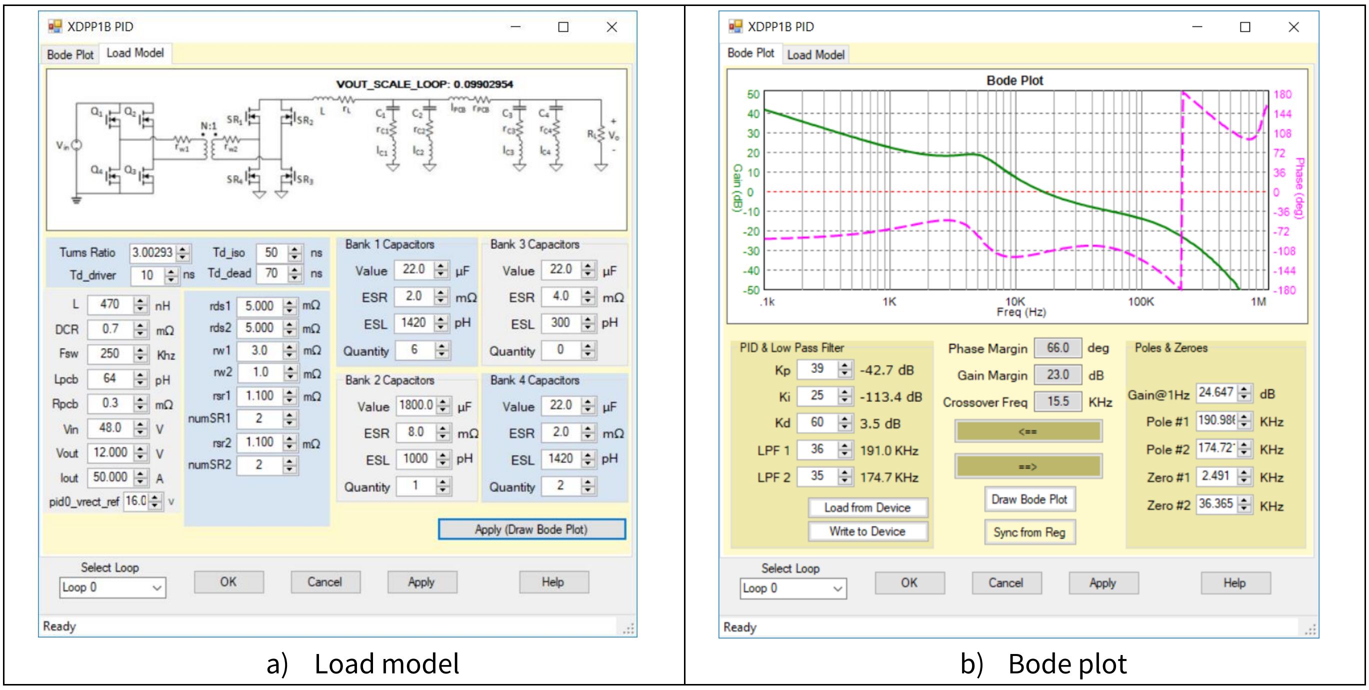
- In VMC the compensator output is clamped to the unsigned range 0.0 to 1.0
- In PCMC the compensator output is "normalized reference current", where the normalization is within the IS ADC maximum range, leading to a signed output range -1.0 to +1.0

The inherent nature of current mode control provides simple dynamics, and therefore, typically for PCMC a type II compensator is sufficient. The difference between type II and III compensators is that type II consists of PI and a single pole whereas type III is PID with two poles. The compensator in **Figure 65** provides a type III response and in order to obtain the type II for PCMC the following need to be considered:

- Compensator includes two single-pole LPFs but neither of them is possible to bypass. However, by setting one of the filter coefficients via register **pid_kfp1_index_1ph** or **pid_kfp2_index_1ph** to maximum bandwidth, the type II response can be approximated.
- Due to the exponential nature of the PID coefficients, setting register **pid_kd_index_1ph** to zero does not result in a zero-valued K_D. However, when PCMC mode is selected (register **mode_control_loop0/1)** it is possible to override the exponential setting and force K_D to be zero via register **pid_kd_index_1ph**. If a non-zero K_D is desired in PCMC it can be set via **pid_kd_index_1ph** to a non-zero setting.

4.5 GUI design tool for PID compensation

The XDPP1100 GUI provides a simulation tool for the user to predict gain and phase margin based on load model and PID configuration. **Figure 71** shows the example of a FB-FB, VMC converter. **Figure 71** a) is the load model of the converter. The accuracy of the bode plot relies on the accuracy of the load model. Major parasitic parameters are included in the load model. Filling in these parameters based on the actual board design is as accurate as possible.



a) Load model

Turns Ratio 3.00293; Td_iso 50 ns; Td_driver 10 ns; Td_dead 70 ns

L	470	nH
DCR	0.7	mΩ
Fsw	250	Khz
Lpcb	64	pH
Rpcb	0.3	mΩ
Vin	48.0	V
Vout	12.000	V
Iout	50.000	A
pid0_vrect_ref	16.0	v

rds1	5.000	mΩ
rds2	5.000	mΩ
rw1	3.0	mΩ
rw2	1.0	mΩ
rsr1	1.100	mΩ
numSR1	2	
rsr2	1.100	mΩ
numSR2	2	

	Bank 1 Capacitors	Bank 2 Capacitors	Bank 3 Capacitors	Bank 4 Capacitors
Value	22.0 µF	1800.0 µF	22.0 µF	22.0 µF
ESR	2.0 mΩ	8.0 mΩ	4.0 mΩ	2.0 mΩ
ESL	1420 pH	1000 pH	300 pH	1420 pH
Quantity	6	1	0	2

Apply (Draw Bode Plot)

Select Loop: Loop 0 | OK | Cancel | Apply | Help

Ready

b) Bode plot

PID & Low Pass Filter

Kp	39	-42.7 dB
Ki	25	-113.4 dB
Kd	60	3.5 dB
LPF 1	36	191.0 KHz
LPF 2	35	174.7 KHz

Load from Device | Write to Device

Phase Margin	66.0	deg
Gain Margin	23.0	dB
Crossover Freq	15.5	KHz

<== | ==> | Draw Bode Plot | Sync from Reg

Poles & Zeroes

Gain@1Hz	24.647	dB
Pole #1	190.98	KHz
Pole #2	174.72	KHz
Zero #1	2.491	KHz
Zero #2	36.365	KHz

Select Loop: Loop 0 | OK | Cancel | Apply | Help

Ready

Figure 71 GUI design tool – PID configuration and bode plot

In the load model, the following parameters and parasitics are included:

- Transformer turns ratio
- Gate driver and isolator delays
- Output inductor value and DCR
- MOSFETs $R_{DS(on)}$, including both primary/control MOSFETs and secondary/SR MOSFETs. For the SR MOSFET, the number of MOSFET in parallel is also defined
- Output capacitor value, ESR and ESL
- Parasitic inductance and resistance of PCB trace of the secondary
- Input and ouput conditions: V_{IN}, V_{OUT}, I_{OUT}
- **pid_vrect_ref**

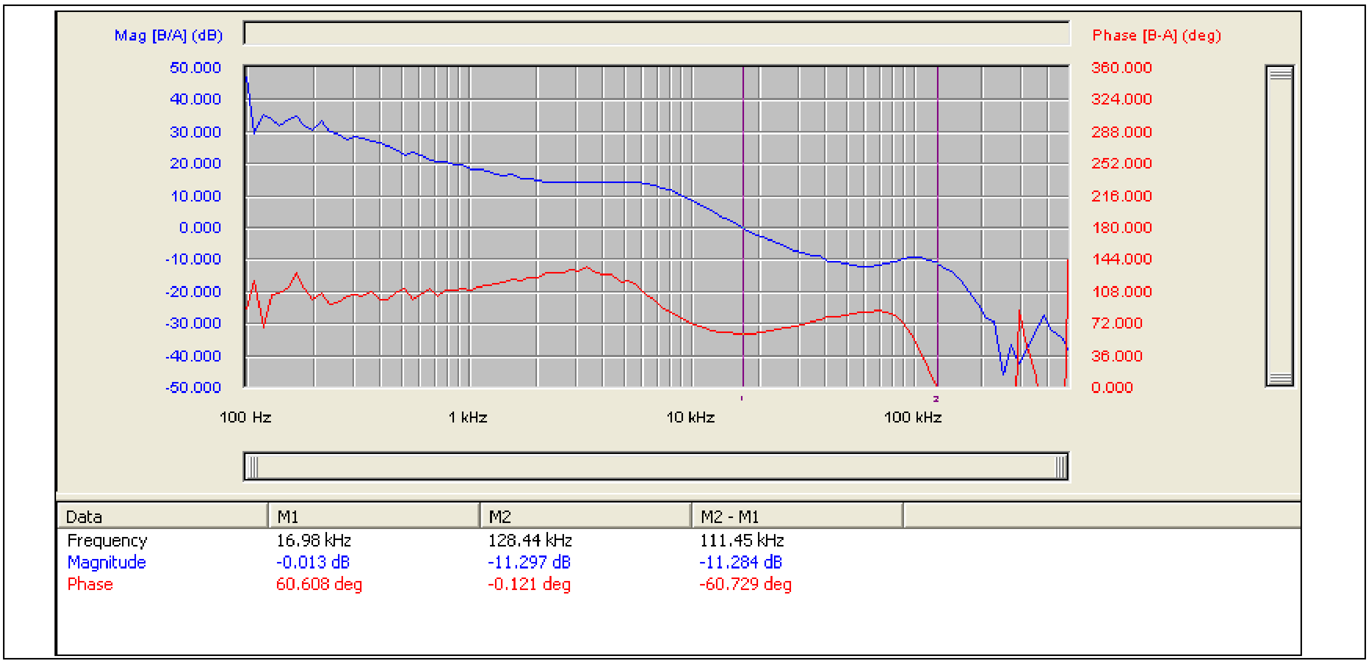
The **pid_vrect_ref** is the PID coefficient scaling reference voltage. PID coefficients are scaled with V_{RECT} to maintain a constant loop gain. This parameter defines the reference V_{RECT} voltage at which the gain scale is 1.0. This parameter should be set to the expected nominal V_{RECT} voltage.

Example: Vin_nom = 48 V, FB topology, turns_ratio = 3:1, pid_vrect_ref = 48 V/3 = 16 V.

Once the load model is defined, go to the bode plot tool to tune the compensation. The tool offers two methods for tuning:

- Manually change Kp, Ki, Kd, LPF1, LPF2 and view the impact on the bode plot in real time
- Use the auto PID function by setting target poles and zeroes and let the tool to calculate and propose the best-matched Kp, Ki, Kd, LPF1, LPF2

In general, the two zeroes can be placed at the double-pole of the output LC filter, and pole 1 can be placed at half of the switching frequency. Click <== to calculate the PID compensation parameters. Once the tool has calculated the PID, click the ==> button to view the adjusted poles and zeroes.



Data	M1	M2	M2 - M1
Frequency	16.98 kHz	128.44 kHz	111.45 kHz
Magnitude	-0.013 dB	-11.297 dB	-11.284 dB
Phase	60.608 deg	-0.121 deg	-60.729 deg

Figure 72 Actual bode plot of the FB-FB converter at 48 V input and 12 V/50 A load

Figure 72 is the actual bode plot of the FB-FB converter. The measured phase margin and crossover frequency are very close to the simulated result.

5 Current sharing

Parallel module connection is often used to deliver higher power in a scalable system. Current sharing between modules should be managed and monitored to avoid overloading a single unit, to distribute power loss and thermal heat evenly, and to achieve overall higher output power. Power supplies with a current sharing function offer more reliable and robust operation in parallel module applications. This chapter discusses passive and active current sharing methods, the XDPP1100 current sharing offerings, and the test results.

5.1 Passive and active current sharing

When multiple power supplies are connected together to provide a common output voltage, current sharing between the units is not guaranteed by default. The power supply that has a higher output voltage would deliver more power to the load than the power supply that has a lower output voltage. The higher-voltage power supply might even source current into the lower voltage power supply if the unit is using synchronous rectification without an ORing stage. Overall system efficiency will be reduced; the power supply that is under higher stress might trigger over-current or overload protection; system shutdown could happen due to over-current or overtemperature.

Current sharing refers to balancing the current of individual power supplies that are connected in parallel. One way of current sharing is passive and uses a resistive load-line. This approach is simple and straightforward. The converter decreases the output voltage with the increasing of output current. The power supply that has higher output initially delivers more current to the load. With load-line, its output voltage drops at high output current, thus reducing sourcing current capability. The unit with a lower voltage would have to supply more current to maintain the total load current. The current between parallel units would eventually reach a balance. The current sharing performance depends on the error output voltage between the parallel units and the load-line impedance. Chapter 5.2 describes the details.

Load-line current sharing can be implemented without any external components or circuit. The downside is that the system's load regulation is degraded. The maximum output power of each unit is reduced due to the reduced output voltage at the maximum rated current. Since the purpose of paralleling modules is to get higher output power, downgraded power capability would be a concern. In this case, non-linear load-line can be considered or use the active current sharing method.

Active current sharing involves using a current sharing wire, which connects all parallel modules together to communicate the average current information between modules. The power supply controller of each module uses this average current information to adjust the output voltage to reduce the error between the module current and the average current. Unlike the passive load-line, which always drops the output voltage at heavy load, active current sharing can adjust the output voltage in both directions. Compared to passive current sharing, active current sharing could achieve higher sharing accuracy without downgrading output power.

The XDPP1100 offers a single wire active current sharing feature. The cost is adding one current sharing resistor to each unit and connecting together the active current sharing pin (IMON) of each converter. Details of the XDPP1100 active current sharing are discussed in chapter 5.3.

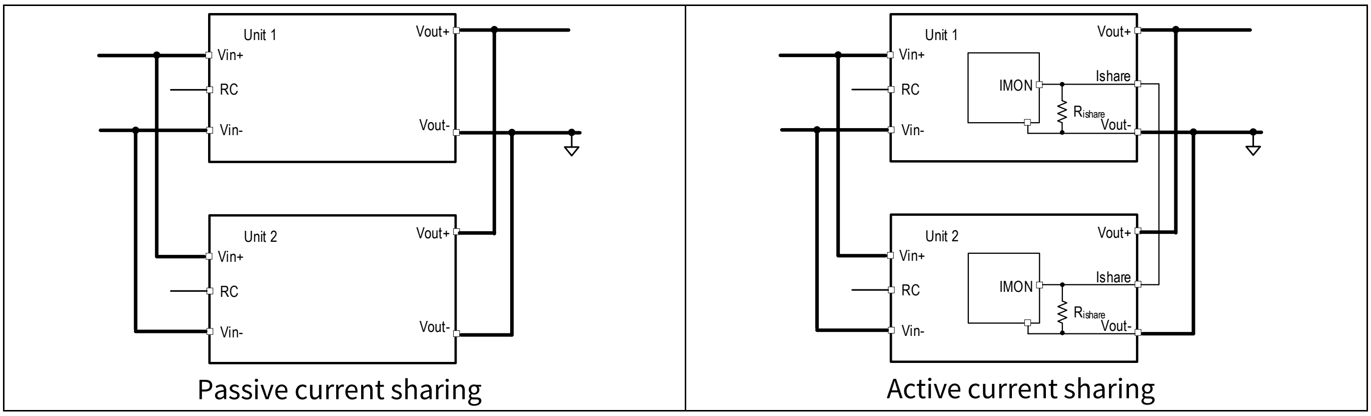
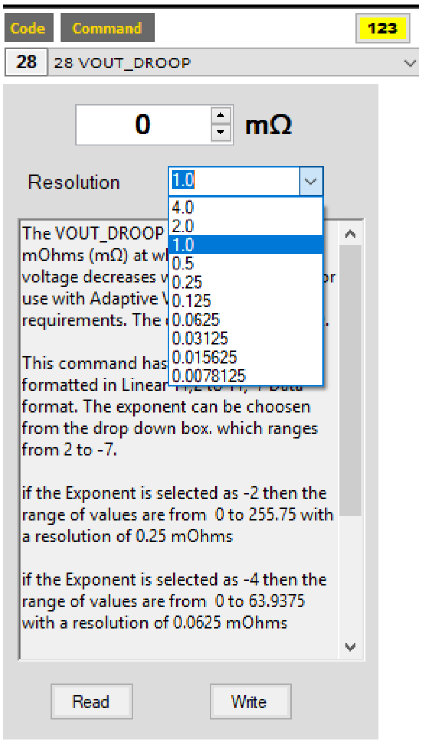


Figure 73 **Current sharing mode**

5.2 Current sharing by droop

5.2.1 VOUT_DROOP

The output impedance of a power supply offers natural load-line, though it usually not sufficient to provide acceptable current sharing performance. A digital power supply is capable of setting configurable voltage droop to achieve better current sharing. For a digital power supply controller that supports PMBus/SMBus, the command VOUT_DROOP defines the output load-line in mΩ. The XDPP1100 supports VOUT_DROOP in LINEAR11 format with the exponent configurable from -7 to +2, which corresponds with resolution from 0.0078125 mΩ to 4 mΩ. **Figure 74** shows the VOUT_DROOP configuration in the XDPP1100 GUI. The resolution can be selected by a drop-down list.



0 mΩ

Resolution 1.0

The VOUT_DROOP
mOhms (mΩ) at wh
voltage decreases w
use with Adaptive V
requirements. The

This command has
formatted in Linear
format. The exponent can be choosen from the drop down box. which ranges from 2 to -7.

if the Exponent is selected as -2 then the range of values are from 0 to 255.75 with a resolution of 0.25 mOhms

if the Exponent is selected as -4 then the range of values are from 0 to 63.9375 with a resolution of 0.0625 mOhms

Figure 74 **VOUT_DROOP command**

VOUT_DROOP should be calculated based on the maximum output current of each unit, and the maximum allowed voltage drop.

- For example:
  - V_{OUT} = 12 V
  - Maximum I_{OUT} = 50 A
  - Acceptable voltage droop = 4 percent, Vout_min at max_Iout = 12 * (1 to 4 percent) = 11.52 V
  - VOUT_DROOP = (12 V * 4 percent)/50 A = 9.6 mΩ

The XDPP1100-Q040 supports two loops. Each loop has the VOUT_DROOP configured independently. Each loop implements the droop calculation based on its own output current value.

Output voltage set-point accuracy plays an important role in load-line current sharing. The more accurate a power supply sets its output voltage, the better the current sharing performance. Take the 12 V/600 W module as an example. If output set-point accuracy is ±1 percent, and two modules are connected in parallel, in the worst case, one unit has output voltage 12.12 V, and the other unit has output voltage 11.88 V. Each unit has VOUT_DROOP set to 10 mΩ. Assume there is no parasitic wire resistance to simplify the estimation. The low-voltage unit will droop voltage to the maximum allowed droop (11.52 V) at 36 A load. The high-voltage unit will droop to 11.52 V at 60 A load. If the rated output current is 50 A, the high-voltage unit would enter the current limit or even shut down due to OCP. The current sharing between the two units is 36 A vs. 60 A.

If the set-point accuracy is ±0.5 percent, then the current sharing between the two units will be 42 A vs. 54 A in the worst case. This is a significant improvement without power stage redesign. A controller with tightly controlled set-point tolerance has the advantage for current sharing.

The output voltage could increase by the droop function if the output current is negative.

5.2.2 Droop clamps

The XDPP1100 provides both positive and negative clamps to output droop. The clamps are independent of the PMBus commands VOUT_MAX and VOUT_MIN. Table 30 lists the clamp registers and design examples. The **vc_vavp_clamp_neg** and **vc_vavp_clamp_pos** set the bounds of the output voltage under droop regulation. If the output voltage reaches the **vc_vavp_clamp_pos** set threshold, further increasing of output current won’t decrease V_{OUT}, and vice versa for the negative droop clamp.

Table 30 Droop clamps

Register name	Description

vc_vavp_clamp_neg	Negative droop clamp voltage can be used to limit negative droop voltage independent of VOUT_MAX (e.g., set to 0 to disable negative droop). Negative droop refers to increasing voltage with negative I_{OUT}.
	LSB = -20 mV
	Range = 0 to -2540 mV

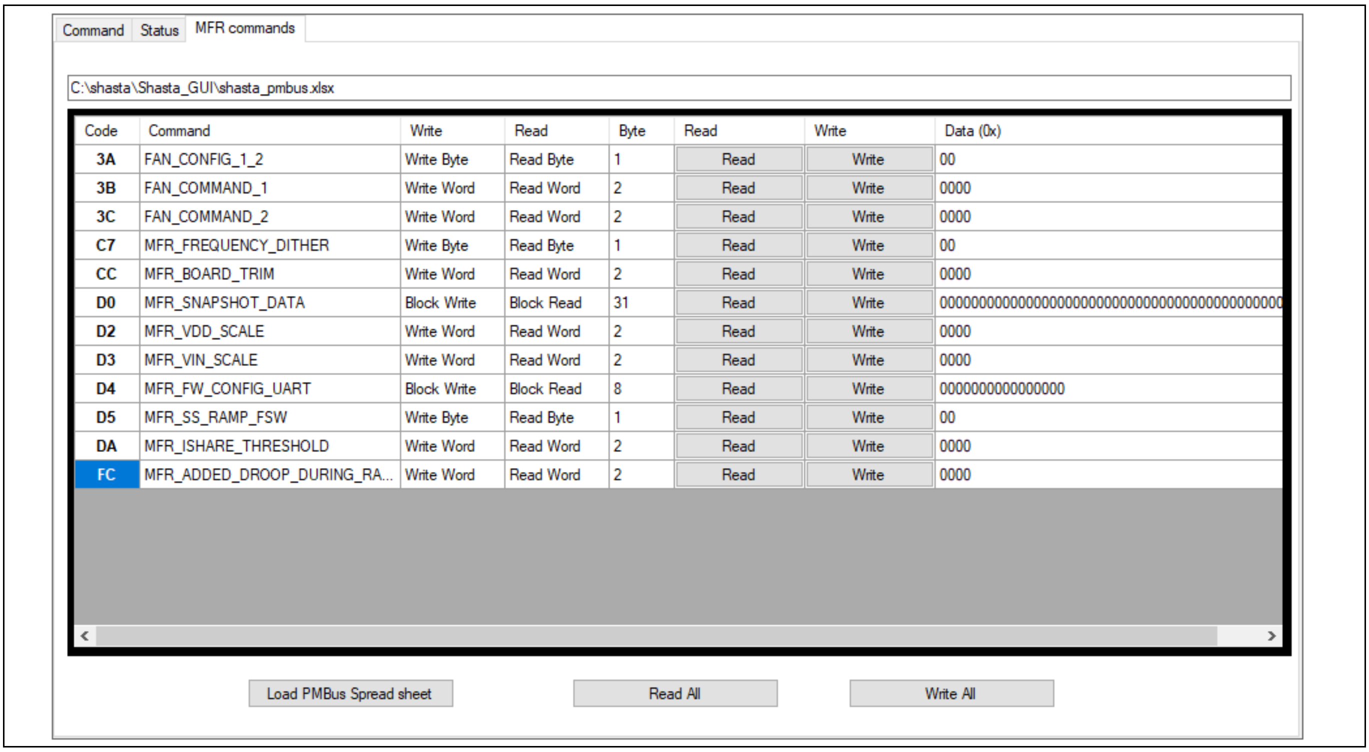
vc_vavp_clamp_pos	Positive droop clamp voltage can be used to limit positive droop voltage independent of VOUT_MIN. Positive droop refers to decreasing voltage with positive I_{OUT}. LSB = 20 mV Range = 0 to 2540 mV • Example: – V_{OUT} = 12 V, positive droop clamp to 11.52 V – Positive clamp voltage = 12 V to 11.52 V = 0.48 V

Register name	Description
	– vc_vavp_clamp_pos = (12 to 11.52 V)/20 mV = 24

5.2.3 ADDED_DROOP_ DURING_RAMP

Current sharing during start-up is always more challenging than in steady-state. The power supplies could have different start-up delays or different ramp times due to device variation. The voltage difference between units could be much more than the set-point error in steady-state.

The XDPP1100 supports adding additional droop during the start-up ramp. The added_droop helps to reduce the error of current sharing during start-up. It is removed once the power supply reaches regulation, thus the max. output power won't be sacrificed. This function is implemented in a XDPP1100 FW patch and could be configured by patched PMBus command 0xFC MFR_ ADDED_DROOP_ DURING_RAMP. Use the "Load PMBus Spread sheet" button to import the MFR PMBus command into the GUI (**Figure 75**). MFR_ ADDED_DROOP_ DURING_RAMP has LINEAR11 data format, and the exponent is configurable from -7 to +2, units in mΩ.



C:\shasta\Shasta_GUI\shasta_pmbus.xlsx

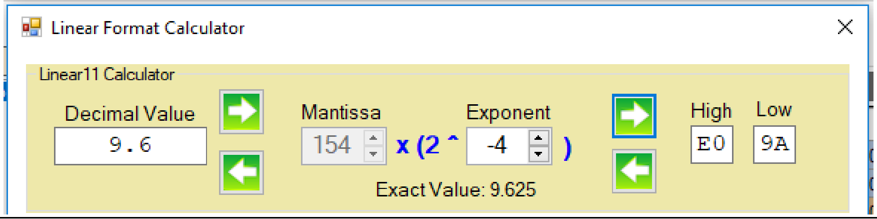
Code	Command	Write	Read	Byte	Read	Write	Data (0x)
3A	FAN_CONFIG_1_2	Write Byte	Read Byte	1	Read	Write	00
3B	FAN_COMMAND_1	Write Word	Read Word	2	Read	Write	0000
3C	FAN_COMMAND_2	Write Word	Read Word	2	Read	Write	0000
C7	MFR_FREQUENCY_DITHER	Write Byte	Read Byte	1	Read	Write	00
CC	MFR_BOARD_TRIM	Write Word	Read Word	2	Read	Write	0000
D0	MFR_SNAPSHOT_DATA	Block Write	Block Read	31	Read	Write	00000000000000000000000000000000000000000000
D2	MFR_VDD_SCALE	Write Word	Read Word	2	Read	Write	0000
D3	MFR_VIN_SCALE	Write Word	Read Word	2	Read	Write	0000
D4	MFR_FW_CONFIG_UART	Block Write	Block Read	8	Read	Write	0000000000000000
D5	MFR_SS_RAMP_FSW	Write Byte	Read Byte	1	Read	Write	00
DA	MFR_ISHARE_THRESHOLD	Write Word	Read Word	2	Read	Write	0000
FC	MFR_ADDED_DROOP_DURING_RA...	Write Word	Read Word	2	Read	Write	0000

Figure 75 MFR_ ADDED_DROOP_ DURING_RAMP command

Unlike the regular droop command (VOUT_DROOP), MFR_ ADDED_DROOP_ DURING_RAMP needs to be written in LINEAR11 format in hex. To help the user convert decimal parameters into LINEAR11, the XDPP1100 GUI offers a linear format calculator (**Figure 76**). For example, to set MFR_ ADDED_DROOP_ DURING_RAMP to 9.6 mΩ:

- Open the calculator $X= Y{\cdot}2^N$.
- Type 9.6 in the "Decimal Value" section.
- Click the right arrow next to the "Decimal Value" box.
- Choose the desired exponent (range -7 to +2); the "Exact Value" would show the actual decimal number based on the selected exponent.

- Click the right arrow next to the “Exponent” box.
- LINEAR11 format data (E0 9A) will show in the right-hand boxes.
- Write E09A to command 0xFC to set MFR_ ADDED_DROOP_ DURING_RAMP to 9.625 mΩ.


Linear Format Calculator

Linear11 Calculator

Decimal Value: 9.6

Mantissa: 154 x (2 ^ Exponent: -4)

High: E0 Low: 9A

Exact Value: 9.625

Figure 76 GUI linear format calculator

5.2.4 Pre-bias start-up with fixed ramp slope

In case a power supply turns on before other parallel units, the one that starts later will start-up in pre-bias condition. To maintain good current sharing in pre-bias start-up, the turn-on ramp slope should be the same for all units. The units that start-up in pre-bias would have a shorter turn-on rise time, which is inversely proportional to pre-bias voltage (**Figure 77**). The XDPP1100 FW calculates T_{ON} rise time based on pre-bias voltage, VOUT_COMMAND and TON_RISE.

$$Ton = \frac{TON\_RISE \times (VOUT\_COMMAND - VBIAS)}{VOUT\_COMMAND} \tag{5.1}$$

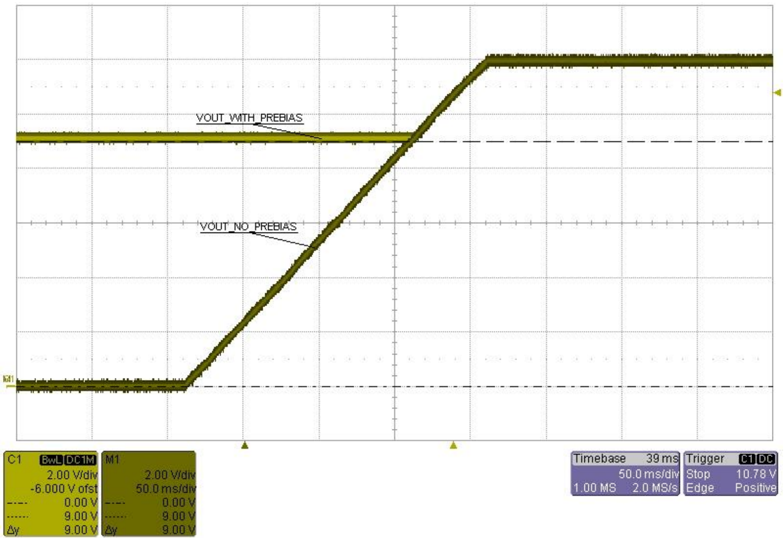


Figure 77 Pre-bias start-up with fixed ramp (C1: pre-bias start-up, M1: 0 V start-up)

5.2.5 Multi-segment droop

Multi-segment droop is a type of non-linear droop that allows the user to define different droop resistances under different load currents. If the load-line resistor is set high, the output voltage will sag quickly when the current exceeds the set threshold; behavior similarly to constant current (CC) or constant power (CP) operation.

Figure 78 shows the behavior of the multi-segment load-line. A load-line value of $R_{LL,neg}$ is used when the output current I_{OUT} is less than zero. This is meant to help current balancing. $R_{LL,1}$ is the regular load-line (VOUT_DROOP) used from 0 A to threshold I_{thr_seg2}. From I_{thr_seg2} to I_{thr_seg3}, $R_{LL,2}$ is used to emulate CC operation. From I_{thr_seg3} until the over-current shutdown threshold, $R_{LL,3}$ is used for approximate constant power operation. $I_{OC,SD}$ equals the output over-current fault threshold IOUT_OC_FAULT_LIMIT.

Note that **Figure 78** demonstrates the CC operation in segment 2. The output voltage slope won't be that sharp if $R_{LL,2}$ is set to a smaller value for current sharing purposes.

The various load-line values can be set to zero to bypass the function.

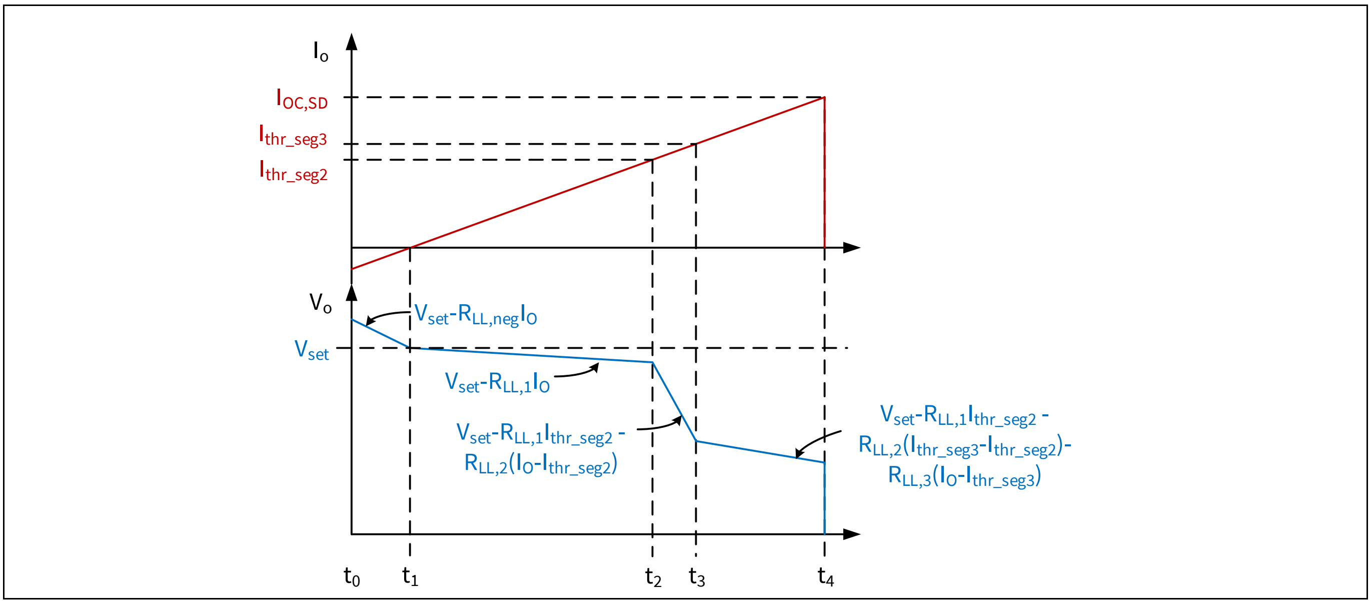
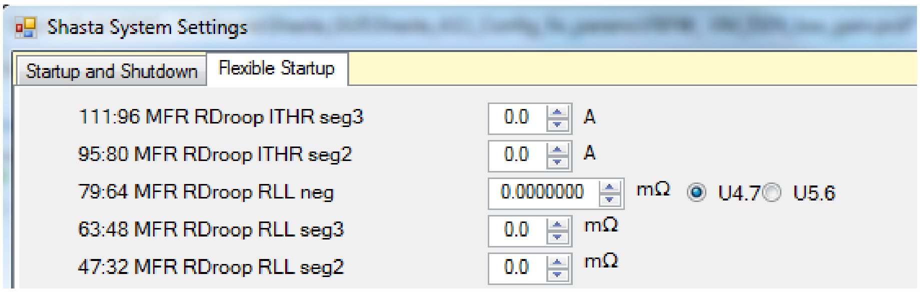


Figure 78 Load-line implementation for OCP

The multi-segment droop parameters are set by PMBus command 0xC5 FW_CONFIG_REGULATION, bit 32:111 (Table 31), or by the GUI design tool "System Setting", on the "Flexible Startup" tab (**Figure 79**).

Table 31 FW_CONFIG_REGULATION data byte format of the multi-segment droop

Bit(s)	Parameter	Meaning	LSB and range
111:96	MFR_RDROOP_ITHR_SEG3	Load-line droop current threshold for third segment to kick in	0.5 A, 255.5 A
95:80	MFR_RDROOP_ITHR_SEG2	Load-line droop current threshold for second segment to kick in	0.5 A, 255.5 A
79:64	MFR_RDROOP_RLL_NEG	Load-line droop for negative segment of three-segment piecewise linear curve	0.0078 mΩ, 15.992 mΩ
63:48	MFR_RDROOP_RLL_SEG3	Load-line droop for third segment of three-segment piecewise linear curve	0.5 mΩ, 511.5 mΩ
47:32	MFR_RDROOP_RLL_SEG2	Load-line droop for second segment of three-segment piecewise linear curve	0.5 mΩ, 511.5 mΩ



111:96 MFR RDroop ITHR seg3 0.0 A

95:80 MFR RDroop ITHR seg2 0.0 A

79:64 MFR RDroop RLL neg 0.0000000 mΩ ● U4.7 ○ U5.6

63:48 MFR RDroop RLL seg3 0.0 mΩ

47:32 MFR RDroop RLL seg2 0.0 mΩ

Figure 79 Multi-segment droop configured by "Flexible Startup" design tool

5.2.6 Droop filters

A LPF **vc_vavp_kfp** is added to VOUT_DROOP to eliminate oscillation during loop regulation. The LPF can be bypassed by setting **vc_vavp_kfp** = $1F_H$ = 31_D. It is recommended to set the bandwidth of **vc_vavp_kfp** lower than the crossover frequency of the feedback loop. The LPF **vc_vavp_kfp** also applies to MFR_ ADDED_DROOP_ DURING_RAMP.

A lower bandwidth LPF **vc_vavp_kfp_lobw** is added to the multi-segment droop, to segment 2 and segment 3. The load-line $R_{LL,2}$ and $R_{LL,3}$ usually have a much higher value than the regular droop when they are used for CC limit and constant power limit. A lower bandwidth is required to avoid oscillation at high gain. This LPF can be bypassed by setting **vc_vavp_kfp_lobw** = $1F_H$ = 31_D.

To help the user determine the LPF value, a design tool in GUI calculates the cut-off frequency for a given number between 0 and 31.

Table 32 Droop filters

Register name	Description
vc_vavp_kfp	LPF coefficient for "high" BW filter applied to the negative and VOUT_DROOP segments, set to all 1s to bypass. • kfp_exp = vavp_kfp [4:2] • kfp_man = 4 + vavp_kfp [1:0] • kfp = kfp_man * 2^(kfp_exp – 14) • F3db (MHz) = [kfp/(1-kfp)] * 25 MHz/2π Range = 0.972 to 195.682 kHz • Example: – vc_vavp_kfp = 18_D = 10010_B – kfp_exp = 4, kfp_man = 4 + 2 = 6 – kfp = 6 * 2^(4 – 14) = 0.005859 – F3db (MHz) = [kfp/(1-kfp)] * 25MHz/2π = 23.451 kHz
vc_vavp_kfp_lobw	LPF coefficient for "low" BW filter applied to the high-gain droop segments 2 and 3, set to all 1s to bypass. • kfp_exp = vavp_kfp_lobw [4:2] • kfp_man = 4 + vavp_kfp_lobw [1:0] • kfp = kfp_man * 2^(kfp_exp – 17) • F3db (MHz) = [kfp/(1-kfp)] * 25 MHz/2π Range = 0.121 to 23.451 kHz

Register name	Description
	• Example: – vc_vavp_kfp = 18_D = 10010_B – kfp_exp = 4, kfp_man = 4 + 2 = 6 – kfp = 6 * 2^(4 – 14) = 0.000732422 – F3db (MHz) = [kfp/(1-kfp)] * 25 MHz/2π = 2.916 kHz

Note: The number with subscript $\#_D$ means the data is in decimal format, with subscript $\#_H$ means the data is in hex format, while subscript $\#_B$ means the data is in binary format.

5.3 XDPP1100 active current sharing

Figure 80 shows the XDPP1100 active current sharing example with two units in parallel.

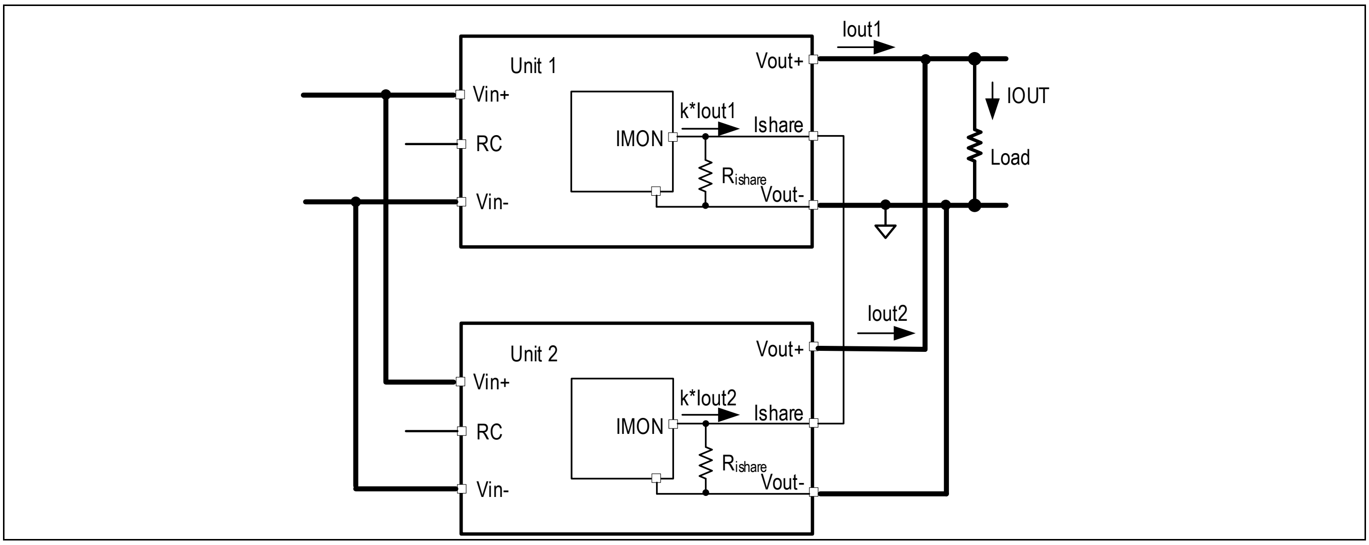


Figure 80 XDPP1100 active current sharing

IMON is an analog DAC output representing the output current. IMON is used for output current monitor, and for active current balancing between multiple parallel modules. A current source proportional to the output current of loop 0 sources out of the IMON pin. The IMON current DAC (IDAC) output range is 0 to 640 µA. The gain of the current source is configurable, which allows the user to scale the current source per application. At no load, this source current is 320 µA. IMON source current lower than 320 µA indicates negative current in this module.

A 1.875 kΩ precision resistor (R_{ishare}) connected between IMON and ground will present a voltage proportional to the output current of each module. At full load, the IMON voltage will be 1.2 V (640 µA x 1.875 kΩ); and at no load, IMON voltage is 0.6 V (320 µA x 1.875 kΩ).

Connecting the IMON of paralleled modules together allows the XDPP1100 to detect the level of average current.

$$IOUT = Iout1 + Iout2 + \cdots + Ioutn \tag{5.2}$$

$$V_{IMON} = (k \cdot Iout1 + k \cdot Iout2 + \cdots + k \cdot Ioutn) \times \frac{R_{ishare}}{n} = k \cdot R_{ishare} \times \frac{IOUT}{n} \tag{5.3}$$

I_{OUT} is the total current supplied to load, *n* is the number of units that are connected in parallel, *k* is the IMON source current scale factor. The voltage of the IMON pin represents the average current.

Each module compares its own output current with the average current and makes the corresponding adjustment. To prevent oscillation on a small error current, a dead zone applies to the current sharing block. When the error current is less than the dead zone, current sharing is inactive.

Figure 81 is the current sharing block diagram of the XDPP1100.

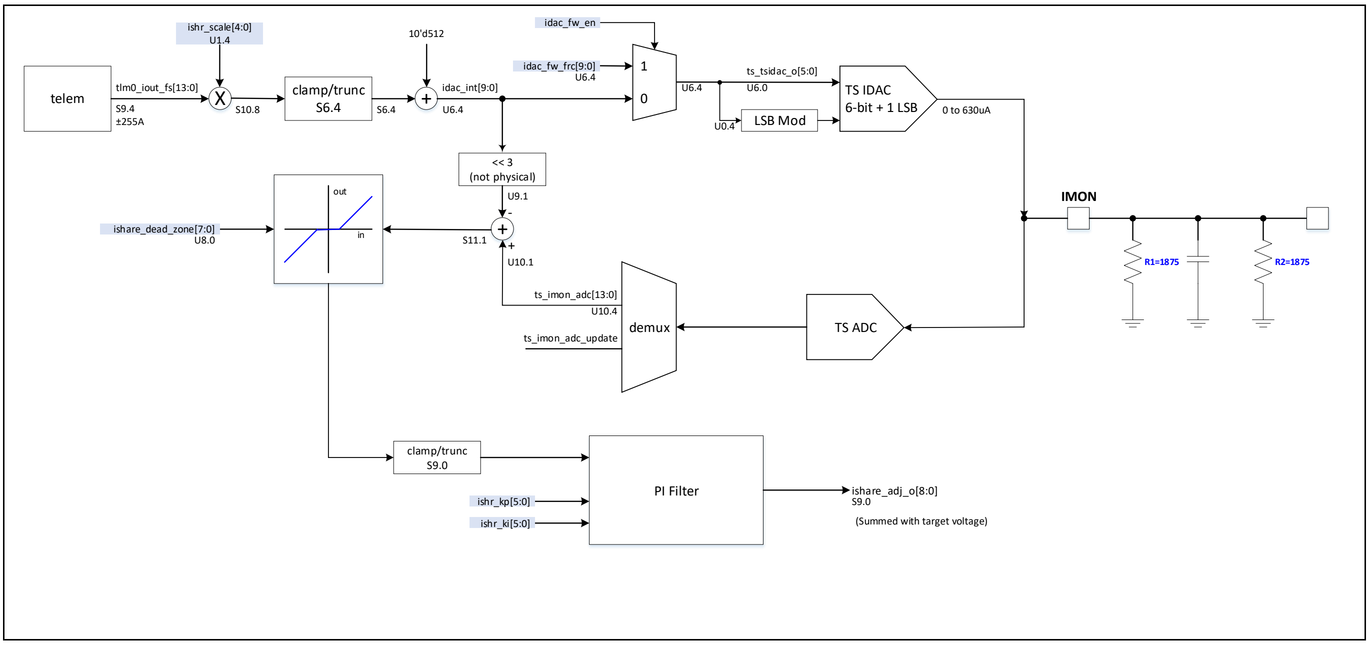


Figure 81 XDPP1100 active current sharing block diagram

- The output current information is taken from the telemetry block; **tlm0_iout_fs** is loop 0 output current cycle averaged at the switching frequency rate.
- Parameter **ishr_scale** scales output current to fit into the DAC range.
- Parameters **idac_fw_en** and **idac_fw_frc** can be used to drive the DAC directly from FW.
- The current telemetry drives a 6-bit current DAC output on the IMON pin.
- Four bits dithering drives an extra LSB input to the DAC for 4-bit extra resolution at the summing node.
- IMON output current from multiple devices is added across R_{ishare} resistor (1.875 kΩ/unit). 1.875 kΩ per device combined with the DAC maximum output of 640 µA will fill the full-scale range of the TS ADC (0 to 1.2 V).
- The TS ADC senses the voltage at IMON as a representation of the average current across all devices.
- The difference between the device current and the average current drives an adjustment to the target output voltage.
- Parameter **ishare_dead_zone** defines an optional dead zone within which the error is forced to 0.

ishr_scale should be calculated based on the maximum current per module:

$$ishr\_scale = integer(16 \times \frac{32}{Iout\_max}) \tag{5.4}$$

Example: The maximum output current per module is 50 A, **ishr_scale** = int (16*32/50) = 10.

If using the example patch that comes with the GUI installation package, the user can configure **ishare_dead_zone** by MFR PMBus command MFR_ISHARE_THRESHOLD. MFR_ISHARE_THRESHOLD defines the error current threshold, below which ISHARE adjustment is zero out. The dead-zone threshold should be set large enough to avoid current sharing failure. For example, if the system READ_IOUT accuracy is +/-1 A, the ISHARE threshold should be larger than 2 A to avoid failure. It should also be set small enough to reduce current sharing error. The FW patch will calculate the **ishare_dead_zone** register based on the following equation:

$$ishare\_dead\_zone = ishare\_scale \times MFR\_ISHARE\_THRESHOLD \qquad (5.5)$$

The PI filter in **Figure 81** consists of a proportional term that works on the instantaneous magnitude of the error, and an integral term that works on the magnitude and the duration of the error. The integral term is the sum of the instantaneous error over time, and it gives the accumulated error. The integral term sets how strongly the loop will response to the “past” information. The integral term sets the low-frequency gain, the proportional term sets the high-frequency gain. The magnitude response of the PI filter is defined by:

$$\sqrt{kp^2 + \left(\frac{ki}{2\pi \cdot Tsw \cdot f}\right)^2} \qquad (5.6)$$

Here, T_{SW} is the converter switching period.

Table 33 Current sharing PI filter register

Register	Description	Equation

ishr_kp	Current sharing PI filter proportional coefficient index. Set to 0 to disable the proportional component of the filter.	kp_exp = ishr_kp [5:3]
		kp_man = 8 + ishr_kp [2:0]
		kp = kp_man * 2^kp_exp * 2^-10

ishr_ki	Current sharing PI filter integral coefficient index. Set to 0 to disable the integral component of the filter.	ki_exp = ishr_ki [5:3]
		ki_man = 8 + ishr_ki [2:0]
		ki = kp_man * 2^ki_exp * 2^-12

Design example:

ishr_kp = 0_D, proportional term is disabled.

ishr_ki = 16_D = 010000_B, ki = 8 * 2^(2-12) = 0.0078125

The PI filter magnitude over-frequency can be plotted as **Figure 82**, at F_{SW} = 250 kHz.

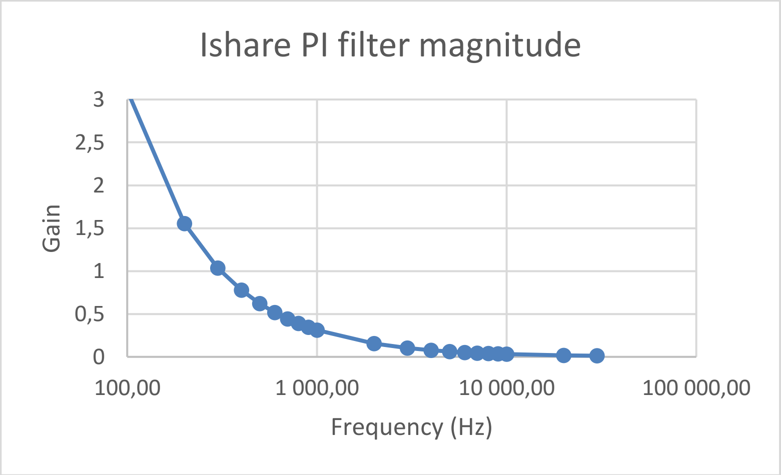


Figure 82 Ishare PI filter example

5.3.1 Active current sharing clamps

The XDPP1100 provides both positive and negative clamps to voltage adjustment under active current sharing. Table 34 lists the clamp registers and design examples.

Table 34 Active current sharing voltage clamps

Register name	Description
ishare_clamp_neg	Negative clamp applied to active current sharing voltage adjustment. This value reflects the voltage at the VSEN input. To convert to V_{OUT} divide by VOUT_SCALE_LOOP. LSB = -1.25 mV Range = 0 to -318.75 mV • Example: – V_{OUT} = 12 V – VOUT_SCALE_LOOP = 0.1 – Clamp V_{OUT} to 11.95 V during active current sharing – Ishare_clamp_neg = (11.95 V-12 V)*0.1/(-1.25 mV) = 4
ishare_clamp_pos	Positive clamp applied to active current sharing voltage adjustment. This value reflects the voltage at the VSEN input. To convert to V_{OUT} divide by VOUT_SCALE_LOOP. LSB = 1.25 mV Range = 0 to 318.75 mV • Example: – V_{OUT} = 12 V – VOUT_SCALE_LOOP = 0.1 – Clamp V_{OUT} to 12.5 V during active current sharing – ishare_clamp_pos = (12.5 V-12 V)*0.1/(1.25 mV) = 40

5.3.2 IMON configuration

Table 35 lists the IMON enable registers.

Table 35 TS ADC IMON enable registers

Register name	Description
imon_meas_en	TS ADC IMON measurement enable. When enabled, the TS ADC will measure the IMON input when selected by ts_muxmode and tx_muxctrl2. When disabled, no IMON measurement will occur, even if selected by ts_muxmode and ts_muxctrl2. Set to 1 for active current sharing.
ts_tsidac_imon_sel	IMON output current DAC enable. This current DAC should be enabled when using the IMON pin for current sharing. The current DAC should be disabled otherwise. Set to 1 for active current sharing.
en_ishare	FW-driven HW block enable for IMON-based current sharing function. 0 = disabled 1 = enabled

Register name	Description
ts_muxmode	Set to 4, 6 or 7 for active current sharing. See chapter 1.5 for TS ADC introduction.

5.3.3 Disconnect R_{ishare} in off mode

In **Figure 80** the R_{ishare} resistor is always connected between IMON and the return/ground of the output. This configuration would give the wrong average current information if one or more parallel units are not in regulation. This could happen for example when the unit has turn-on delay or is shut down due to over-current or overtemperature protection. While the unit is in off-mode, the R_{ishare} resistor is still in parallel with the other R_{ishare} resistors on the IMON bus. This reduces the total equivalent IMON resistance and the XDPP1100 measures average current lower than the actual value.

The R_{ishare} resistor should be disconnected from the IMON bus when the converter is not in operation. PWM11 and PWM6 are the pins that have hot N-WELL output cells. The outputs of PWM11 and PWM6 are in tri-state (HiZ) when the XDPP1100 is not biased. This allows for connecting this pin to a voltage prior to applying V_{DD}. PWM11 and PWM6 can be used as a switch that floats R_{ishare} when the XDPP1100 is not in operation and connects R_{ishare} to ground when the XDPP1100 is enabled.

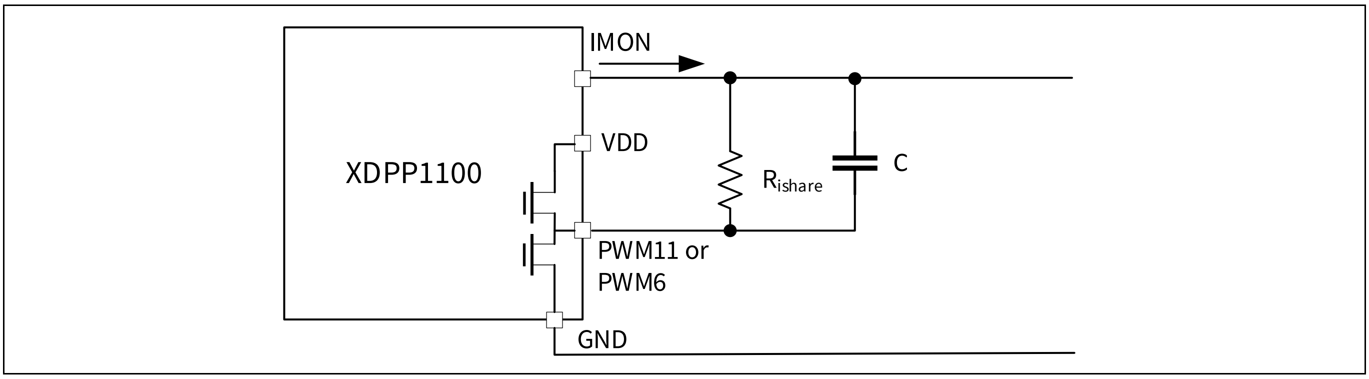


Figure 83 Use PWM11 (or PWM6) to disconnect R_{ishare}

The feature is offered in the XDPP1100 FW patch. Prior to start-up, the IMON DAC is disabled (**ts_tsidac_imon_sel** = 0), and **en_ishare** is disabled (**en_ishare** = 0). The IMON resistor is disconnected from the circuit with the PWM11 output floating. When the converter is enabled, active current sharing will be enabled (**en_ishare** = 1) if the MFR PMBus command 0xDA MFR_ISHARE_THRESHOLD is none zero. The IMON current source is enabled (**ts_tsidac_imon_sel** = 1), and PWM11 will be pulled low to have the IMON resistor connected in the circuit. The active current sharing is enabled at the beginning of the output ramp.

Figure 84 shows PWM11 and IMON waveforms at start-up. In this test, two converters are connected in parallel with the outputs shorted together. Each unit has a 1.87 kΩ IMON resistor connected to its own XDPP1100 PWM11 pin. The IMON wires of the two converters are connected together. Unit #1 has 0 ms turn-on delay and unit #2 has 100 ms turn-on delay. The waveform shows the V_{OUT}, IMON voltage and PWM11 of unit #2. There is no load to the output, so IMON voltage is expected to be 600 mV.

The PWM11 of unit #2 was 0.2 V prior to both units' start-up and was pulled up to 0.6 V by unit #1 through the R_{ishare} resistor when unit #1 started ramping up. It was then pulled down to 0 V when unit #2 started operation. Both the output voltage and IMON bus were smooth during the start-up.

DE mode is enabled at start-up to allow pre-bias condition.

A 1 nF filter cap is recommended to be connected to the IMON pin to filter voltage spikes during **ts_tsidac_imon_sel** and PWM11 transient.

Ch2: IMON voltage, Ch3: PWM11 of unit #2, Ch4: V_{OUT}

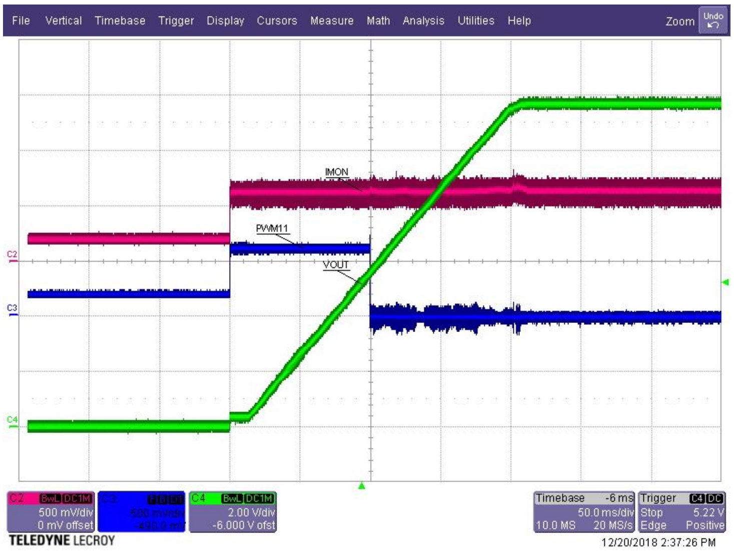


Figure 84 Current sharing start-up waveforms (two units in parallel)

5.3.4 Current share fault

When active current sharing is enabled, the IC monitors the error between the output current and the average current measured at the IMON pin. If the error is more than the MFR_ISHARE_THRESHOLD, the XDPP1100 adjusts the output voltage to reduce the error. If the output voltage adjusts to the active current sharing clamps, and the error is still larger than the MFR_ISHARE_THRESHOLD, "current share fault" is reported. Bit 3 of the STATUS_IOUT is set to 1.

5.4 Current sharing register descriptions

Table 36 describes the registers used by the current sharing function.

Table 36 Current sharing relevant register descriptions

Name	Address (loop 0/1)	Bits	Description
Common peripheral			
imon_func	7000_3008$_H$	[2:0]	Pin IMON function definition. Set to 0 to enable analog IMON function for active current sharing. <table><tr><th>[2:0]</th><th>Direction</th><th>Function</th><th>[2:0]</th><th>Direction</th><th>Function</th></tr><tr><td>0</td><td>Analog</td><td>IMON</td><td>4</td><td>I</td><td>FAN1_TACH</td></tr><tr><td>1</td><td>IO</td><td>GPIO0[3]</td><td>5</td><td>na</td><td>na</td></tr><tr><td>2</td><td>IO</td><td>GPIO1[3]</td><td>6</td><td>na</td><td>na</td></tr><tr><td>3</td><td>IO</td><td>SYNC</td><td>7</td><td>na</td><td>na</td></tr></table>
imon_pd	7000_3008$_H$	[3]	Pin IMON weak pull-down enable. Set to 0 for active current sharing. 0 = pull-down disabled 1 = pull-down enabled
imon_pu_n	7000_3008$_H$	[4]	Pin IMON weak pull-up enable. Set to 1 for active current

Name	Address (loop 0/1)	Bits	Description
			sharing. 0 = pull-up enabled 1 = pull-up disabled
imon_ppen	7000_3008_H	[5]	Pin IMON output buffer CMOS/open drain select. Set to 0 for active current sharing. 0 = open drain output 1 = CMOS output
ishr_scale	7000_3020_H	[4:0]	Used for current sharing, this register defines a pre-scale gain applied to the internal current telemetry before sending to the current output DAC on the IMON pin. Its setting should be computed as follows: ishr_scale(U1.4) = integer(16 * (32/max current))
ishr_kp	7000_3020_H	[10:5]	Current sharing PI filter proportional coefficient index. Set to 0 to disable the proportional component of the filter. Note that index settings greater than 55 are clamped to 55. • kp_exp = ishr_kp [5:3] • kp_man = 8 + ishr_kp [2:0] • kp = kp_man * 2^(kp_exp – 10)
ishr_ki	7000_3020_H	[16:11]	Current sharing PI filter integral coefficient index. Set to 0 to disable the integral component of the filter. Note that index settings greater than 55 are clamped to 55. • ki_exp = ishr_ki [5:3] • ki_man = 8 + ishr_ki [2:0] • ki = ki_man * 2^(ki_exp – 12)
ishare_clamp_neg	7000_3024_H	[7:0]	Negative clamp applied to active current sharing voltage adjustment. This value reflects the voltage at the VSEN input. To convert to V_{OUT} divide by VOUT_SCALE_LOOP. LSB = -1.25 mV, range = 0 to -318.75 mV
ishare_clamp_pos	7000_3024_H	[15:8]	Positive clamp applied to active current sharing voltage adjustment. This value reflects the voltage at the VSEN input. To convert to V_{OUT} divide by VOUT_SCALE_LOOP LSB = 1.25 mV, range = 0 to 318.75 mV
en_ishare	7000_3054_H	[12]	FW-driven HW block enable for IMON-based current sharing function. 0 = disabled 1 = enabled
ishr_fw_adj	7000_3084_H	[11:0]	When ishr_fw_en is high, this register overrides the HW current share voltage adjust output with a FW controlled setting.
ishr_fw_en	7000_3084_H	[12]	Enables FW controlled current share loop via ishr_fw_adj. 0 = use HW computed current share adjust 1 = use ishr_fw_adj

Name	Address (loop 0/1)	Bits	Description
ishare_dead_zone	7000_30A0_H	[7:0]	ISHARE dead zone below which current sharing is not attempted. To convert to amps divide by ishr_scale. This register will be calculated by FW based on PMBus command MFR_ISHARE_THRESHOLD. LSB = 1 ADC code, range = 0 to 255 • Example: – ishr_scale = 10 – MFR_ISHARE_THRESHOLD = 3 (A) – ishare_dead_zone= 3 * 10 = 30
pwm6_func	7000_3010_H	[2:0]	Pin PWM6 function definition. Set to 1 to configure as GPIO output for R_{ishare} control.
pwm6_pd	7000_3010_H	[3]	Pin PWM6 weak pull-down enable. 0 = pull-down disabled (recommended) 1 = pull-down enabled
pwm6_pu_n	7000_3010_H	[4]	Pin PWM6 weak pull-up enable. 0 = pull-up enabled 1 = pull-up disabled (recommended)
pwm6_ppen	7000_3010_H	[5]	Pin PWM6 output buffer CMOS/open drain select. Set to 0 when using PWM6 as the R_{ishare} GND switch in active current sharing. 0 = open drain output 1 = CMOS output
pwm6_static_hiz	7000_3010_H	[30]	Pin PWM6 static HiZ control. Pin PWM6 has a special output buffer with tri-state bias resistors for use with integrated power stage drivers or to control active current sharing resistors. Set to 0 for R_{ishare} control. 0 = tri-state biasing disabled (typical usage) 1 = tri-state biasing enabled (integrated power stage usage)
pwm11_func	7000_3014_H	[2:0]	Pin PWM11 function definition. Set to 1 to configure as GPIO output for R_{ishare} control.
pwm11_pd	7000_3014_H	[3]	Pin PWM11 weak pull-down enable. 0 = pull-down disabled (recommended) 1 = pull-down enabled
pwm11_pu_n	7000_3014_H	[4]	Pin PWM11 weak pull-up enable. 0 = pull-up enabled 1 = pull-up disabled (recommended)
pwm11_ppen	7000_3014_H	[5]	Pin PWM11 output buffer CMOS/open drain select. Set to 0 when using PWM11 as the R_{ishare} GND switch in active current sharing. 0 = open drain output 1 = CMOS output

Name	Address (loop 0/1)	Bits	Description
pwm11_static_hiz	7000_3014$_H$	[12]	Pin PWM11 static HiZ control. Pin PWM11 has a special output buffer with tri-state bias resistors for use with integrated power stage drivers or to control active current sharing resistor. Set to 0 for R_{ishare} control. 0 = tri-state biasing disabled (typical usage) 1 = tri-state biasing enabled (integrated power stage usage)
TSEN peripheral			
imon_meas_en	7000_4C00$_H$	[1]	TS ADC IMON measurement enable. When enabled, the TS ADC will measure the IMON input when selected by ts_muxmode and tx_muxctrl2. When disabled, no IMON measurement will occur, even if selected by ts_muxmode and ts_muxctrl2. 0 = disabled 1 = enabled
ts_tsidac_imon_sel	7000_4C00$_H$	[8]	IMON output current DAC enable. This current DAC should be enabled when using the IMON pin for current sharing. The current DAC should be disabled otherwise. 0 = disabled 1 = enabled
ts_muxctrl1	7000_4C00$_H$	[16:15]	TS ADC MUX1 input source select. The output of MUX1 is connected to MUX2 input 7. The most common setting of this register is 0 to measure the internal temperature of the controller.
ts_muxctrl2	7000_4C00$_H$	[19:17]	TS ADC MUX2 input source select. The output of MUX2 is connected to the TS ADC input. Generally, this mux is auto-sequenced by the TS ADC but may be overridden by setting parameter ts_tsmuxmode = 0.
ts_muxmode	7000_4C00$_H$	[22:20]	TS ADC input sequence control. When bit [2] is 0, the TS ADC input is entirely determined by the settings of ts_muxctrl1 and ts_muxctrl2. When bit [2] is 1, MUX2 auto-sequences its input using the pattern in the table below. If the sequences include MUX2 input 7 (MUX1), the source in this timeslot is determined by the setting of ts_muxctrl1.
$V_{control}$ peripheral			
vc_vavp_clamp_neg	7000_1400$_H$ 7000_1800$_H$	[6:0]	Negative droop (load-line) clamp voltage, can be used to limit negative droop voltage independent of VOUT_MAX (e.g., set to 0 to disable negative droop). Note: Positive droop refers to decreasing voltage with positive I_{OUT}, negative droop refers to increasing voltage with negative I_{OUT}. LSB = -20 mV, range = 0 to -2540 mV
vc_vavp_clamp_pos	7000_1400$_H$ 7000_1800$_H$	[13:7]	Positive droop (load-line) clamp voltage, can be used to limit positive droop voltage independent of VOUT_MIN. Note: Positive droop refers to decreasing voltage with

Name	Address (loop 0/1)	Bits	Description
			positive I_{OUT}, negative droop refers to increasing voltage with negative I_{OUT}. LSB = 20 mV, range = 0 to 2540 mV
vc_vavp_kfp	7000_1400_H 7000_1800_H	[18:14]	LPF coefficient for "high" BW filter applied to the negative and VOUT_DROOP segments, set to all 1s to bypass. • kfp_exp = vavp_kfp [4:2] • kfp_man = 4 + vavp_kfp [1:0] • kfp = kfp_man * 2^(kfp_exp – 14) • F3db (MHz) = [kfp/(1-kfp)] * 25 MHz/2π Range = 0.972 to 195.682 kHz
vc_vavp_kfp_lobw	7000_1400_H 7000_1800_H	[23:19]	LPF coefficient for "low" BW filter applied to the high-gain droop segments 2 and 3, set to all 1s to bypass. • kfp_exp = vavp_kfp_lobw [4:2] • kfp_man = 4 + vavp_kfp_lobw [1:0] • kfp = kfp_man * 2^(kfp_exp – 17) • F3db (MHz) = [kfp/(1-kfp)] * 25 MHz/2π Range = 0.121 to 23.451 kHz

5.5 PMBus command descriptions

Table 37 describes the PMBus commands relevant to the current sharing function.

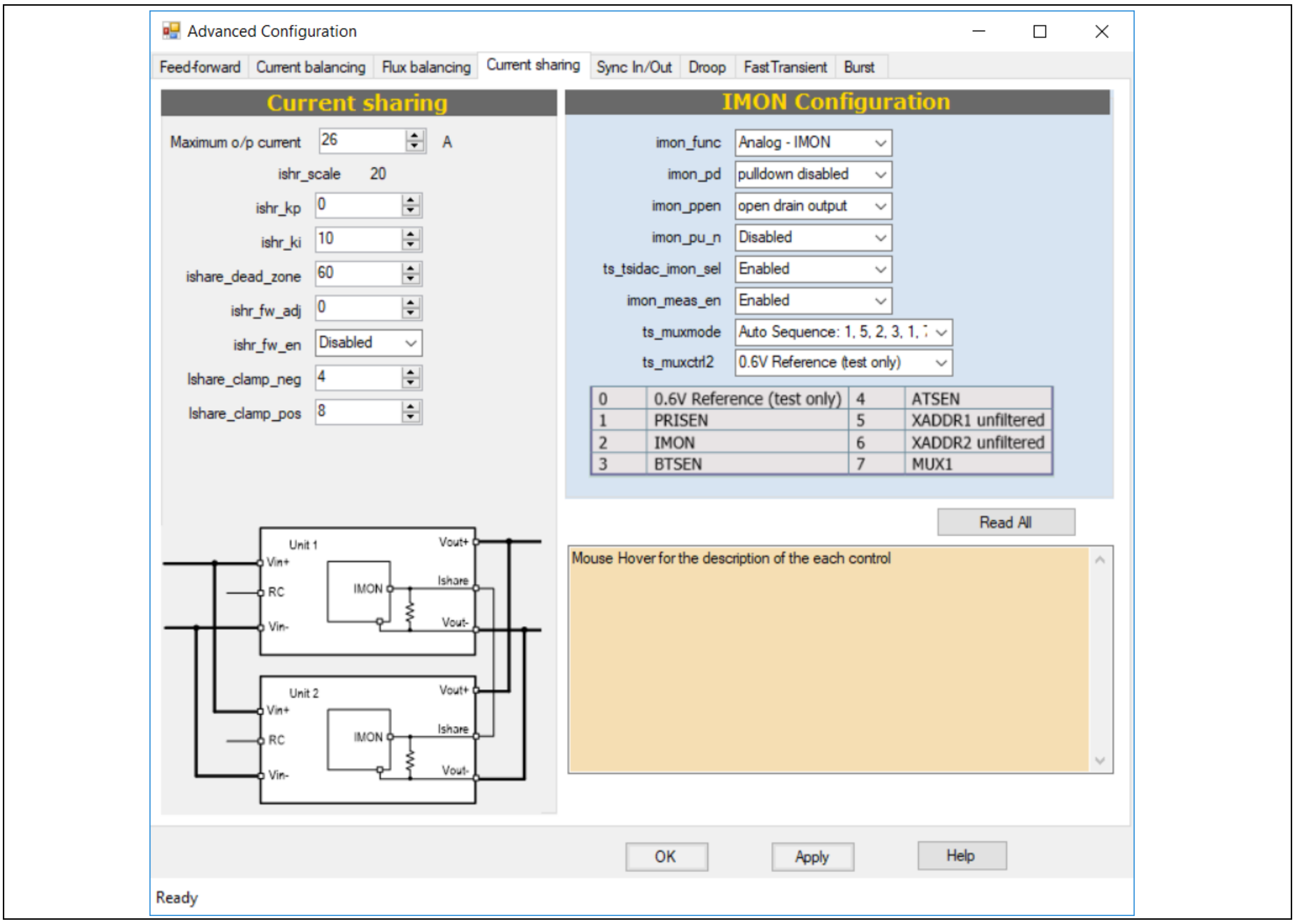
Table 37 Current sharing relevant PMBus command descriptions

Command name	Command code	Format	Description
VOUT_DROOP	28_H	LINEAR11	The VOUT_DROOP sets the rate, in mV/A (mΩ) at which the output voltage decreases (or increases) with increasing (or decreasing) output current for use with adaptive voltage positioning requirements and passive current sharing schemes.
IOUT_CAL_OFFSET	39_H	LINEAR11	Defines positive or negative offset added to current sense that calibrates and nulls out any offsets. Unit: amps
MFR_IOUT_APC	EA_H	LINEAR11	Defines current sense gain. Unit: amps/code
MFR_ISHARE_THRESHOLD	DA_H	LINEAR11	Defines current sharing dead-zone. This is a patched MFR PMBus command. It is valid only if the function is defined in the FW patch. Unit: amps
MFR_ADDED_DROOP_ DURING_RAMP	FC_H	LINEAR11	Defines added droop during start-up ramp. This is a patched MFR PMBus command. It is valid only if the function is defined in the FW patch.

Command name	Command code	Format	Description
			Unit: mΩ
FW_CONFIG_REGULATION	C5$_H$		See “Multi-segment droop” section 5.2.5.

5.6 XDPP1100 GUI design tool for current sharing configuration

The XDPP1100 GUI design tool for active current sharing is shown in **Figure 85**.



Maximum o/p current 26 A
ishr_scale 20
ishr_kp 0
ishr_ki 10
ishare_dead_zone 60
ishr_fw_adj 0
ishr_fw_en Disabled
Ishare_clamp_neg 4
Ishare_clamp_pos 8

imon_func Analog - IMON
imon_pd pulldown disabled
imon_ppen open drain output
imon_pu_n Disabled
ts_tsidac_imon_sel Enabled
imon_meas_en Enabled
ts_muxmode Auto Sequence: 1, 5, 2, 3, 1, 7
ts_muxctrl2 0.6V Reference (test only)

0	0.6V Reference (test only)	4	ATSEN
1	PRISEN	5	XADDR1 unfiltered
2	IMON	6	XADDR2 unfiltered
3	BTSEN	7	MUX1

Mouse Hover for the description of the each control

Figure 85 GUI design tool

5.7 Test result of current sharing

In this section, current sharing is tested with a 600 W 48 V-to-12 V quarter-brick reference design. The document of the reference design is available online: **REF_600W_FBFB_XDPP1100**. A FW patch is loaded to the XDPP1100 OTP for active current sharing control. The source code of the patch can be found in XDPP1100 GUI installation folder C:\XDPP1100\XDPP1100_fw\projects\patch_user_app.

The three-unit quarter-/eighth-brick test fixture enables connecting three quarter-brick units in parallel. Each unit has its individual on/off control switch. There is also a main switch that can turn-on all the units at the same time.

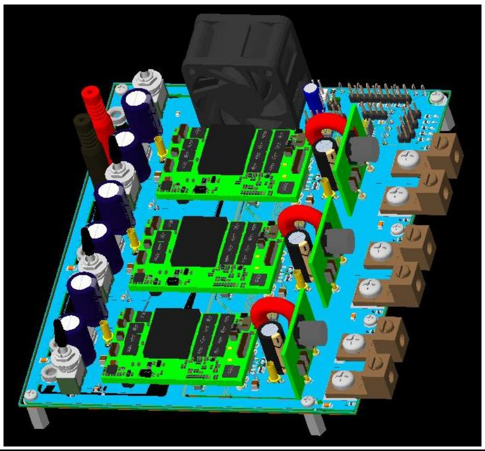


Figure 86 Three-unit QB test fixture with three parallel units

5.7.1 Active current sharing configuration

Table 38 shows the registers that should be configured by the user for active current sharing. The board has VOUT_SCALE_LOOP = 0.099; this scale is used when calculating the output voltage clamp thresholds.

Table 38 Active current sharing registers configured by user

Register	Value	Description
imon_func	0	Analog IMON function is selected
imon_pd	0	IMON pin pull-down disable
imon_pu_n	1	IMON pin pull-up disable
imon_ppen	0	IMON pin output buffer set to open drain
ishr_scale	10	Integer (16 * (32/50 A)) = 10
ishr_kp	0	Proportional term of the PI filter is disabled
ishr_ki	8	Integral coefficient of the PI filter
ishare_clamp_neg	2	Negative voltage is clamped to 11.975 V during active current sharing
ishare_clamp_pos	10	Positive voltage is clamped to 12.125 V during active current sharing
ishr_fw_en	0	Use HW computed current share adjustment
imon_meas_en	1	TS ADC IMON measurement enabled
ts_muxmode	7	TS ADC input auto sequence, measure IMON input once every 32 µs
pwm11_pd	0	PWM11 pull-down disable
pwm11_pu_n	1	PWM11 pull-up disable
pwm11_ppen	0	PWM11 output buffer set to open drain
pwm11_static_hiz	0	PWM11 tri-state biasing disabled

MFR PMBus command 0xDA MFR_ISHARE_THRESHOLD sets current sharing dead-zone. Setting it to a non-zero value also triggers FW-controlled current sharing features. The MFR_ISHARE_THRESHOLD should be set to at least twice the accuracy of I_{OUT} telemetry to avoid voltage adjustment going in the wrong direction. In this test, MFR_ISHARE_THRESHOLD is set to 2.5 A.

Table 39 PMBus command for active current sharing

PMBUS command	Value	Description
VOUT_SCALE_LOOP	0.099	Output voltage sense resistor divider ratio is 0.099
MFR_ISHARE_THRESHOLD	F805$_H$	Current sharing dead-zone is 2.5 A

Table 40 shows the registers controlled by the FW.

Table 40 Active current sharing registers controlled by FW

Register	Off mode/fault	Operation
en_ishare	0	1
ishare_dead_zone	= ishr_scale * MFR_ISHARE_THRESHOLD = 22	
ts_tsidac_imon_sel	0	1
Pwm11_func	1 (GPIO)	1 (GPIO)
Pwm11_ppen	0 (open drain)	1 (CMOS, pulled down)

5.7.2 Start-up comparison: DROOP vs. active current sharing

To verify the current sharing performance, the V_{OUT} of the two modules was intentionally set to have 100 mV voltage difference.

Unit #1, VOUT_COMMAND = 11.95 V, xaddr 0x41

Unit #2, VOUT_COMMAND = 12.05 V, xaddr 0x40

On the test fixture, remove the remote sense resistor of each unit to enable local voltage sensing.

Test case 1: active current sharing is disabled, current sharing by droop only:

- Added droop is 14 mΩ at start-up ramp and reduced to 9 mΩ at regulation
- vc0_vavp_kfp = 24 (output voltage sense LPF)
- vc0_vavp_clamp_neg = 0 (negative droop clamp)
- vc0_vavp_clamp_pos = 24 (positive droop clamp, -480 mV at V_{OUT})
- Enable DE start-up of one unit (can be either unit #1 or unit #2), the waveform is taken with unit #2 DE start-up enabled

Test case 2: active current sharing is enabled:

- No added droop at start-up ramp, standard droop 4 mΩ at regulation
- vc0_vavp_kfp = 24
- ishr_scale = 10 (current share scale, defined per the maximum output current 50 A)
- ishr_ki = 8, ishr_kp = 0 (active current share PI filter)
- ishare_clamp_neg = 2 (negative clamp, -25 mV at V_{OUT})
- ishare_clamp_pos = 10 (positive clamp, +125 mV at V_{OUT})
- MFR_ISHARE_THRESHOLD = 0x0002 (2 A)
- Enable unit #2 DE start-up

Ch1: IOUT1 of unit#1 (5 A/div), Ch2: V_{OUT} (2 V/div), Ch3: IMON pin (0.5 V/div), Ch4: IOUT2 of unit#2 (5 A/div)

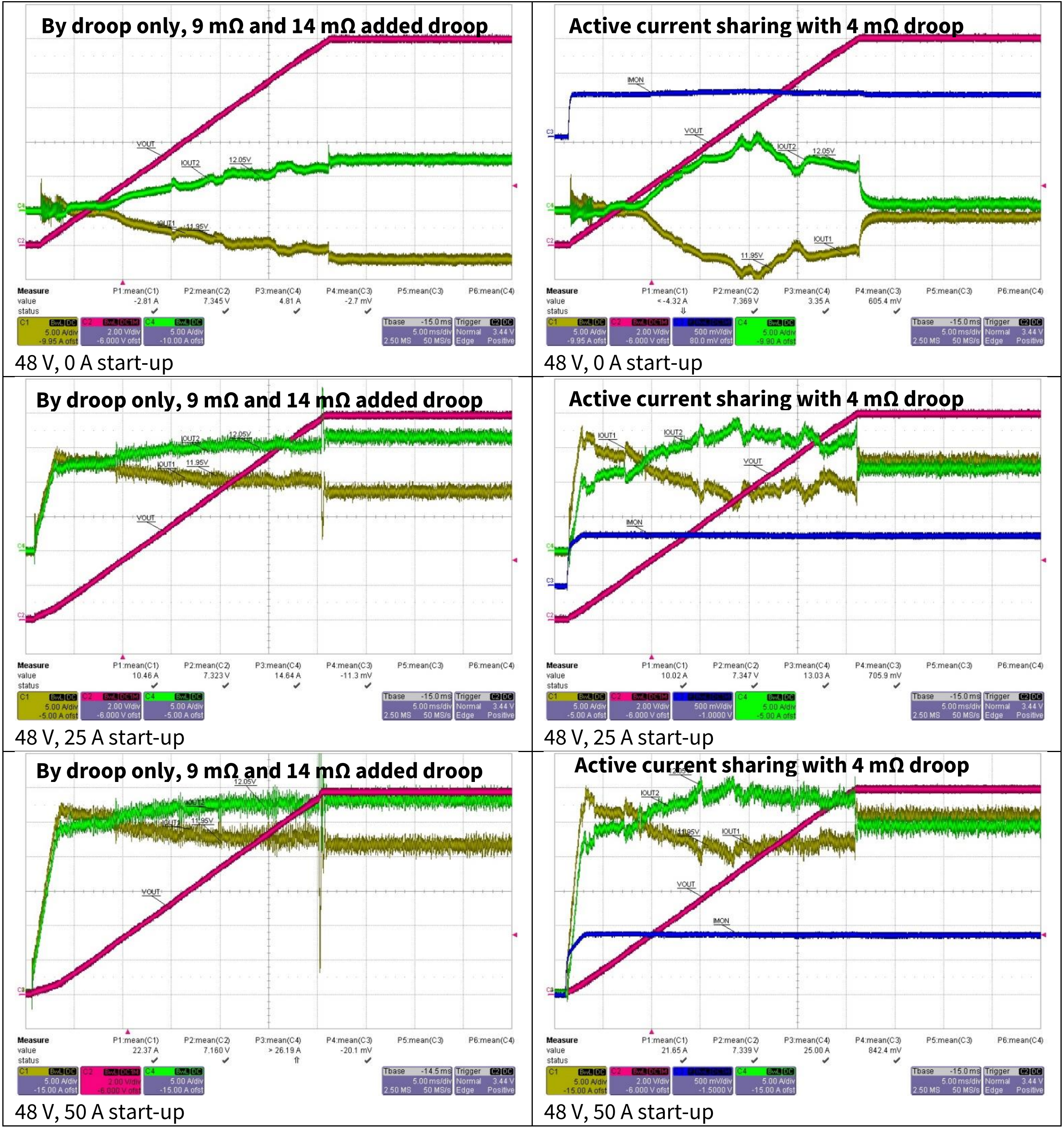


Figure 87 Current sharing waveform at start-up

In steady-state, the active current sharing has much better performance than the droop-based current sharing. The active current sharing at start-up, however, gets larger errors, especially at no load. This can be improved by adding some droop during the start-up ramp. Also, if the unit could sense negative output current, it would improve the current sharing accuracy at no load.

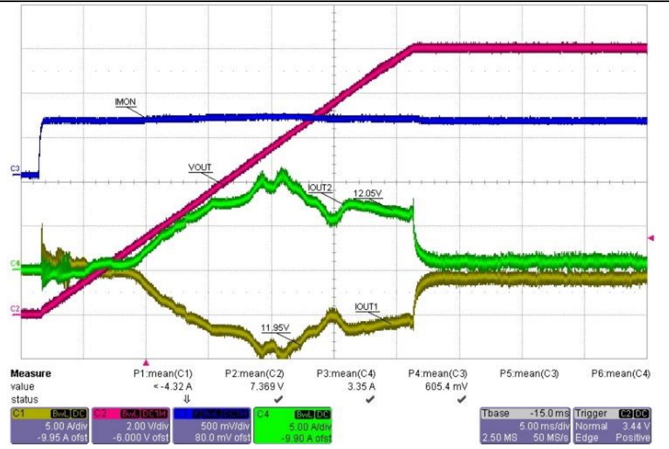
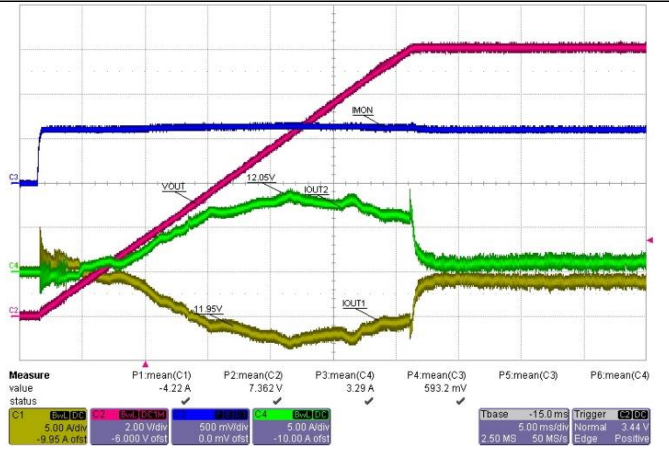
Active current sharing with 4 mΩ droop	Active current sharing with 4 mΩ droop and 6 mΩ added droop at ramp
 48 V, 0 A start-up	 48 V, 0 A start-up

Figure 88 Start-up waveforms with active current sharing

5.7.3 Start-up with different TON_DELAY

The test is done with two units in parallel. One of the units has 10 ms TON_DELA,Y which makes the unit start-up in pre-bias condition. TON_RISE of both units are set to the same (30 ms).

Unit #1, VOUT_COMMAND = 11.95 V, xaddr 0x41, TON_DELAY = 0, TON_RISE = 30 ms
Unit #2, VOUT_COMMAND = 12.05 V, xaddr 0x40, TON_DELAY = 10 ms, TON_RISE = 30 ms

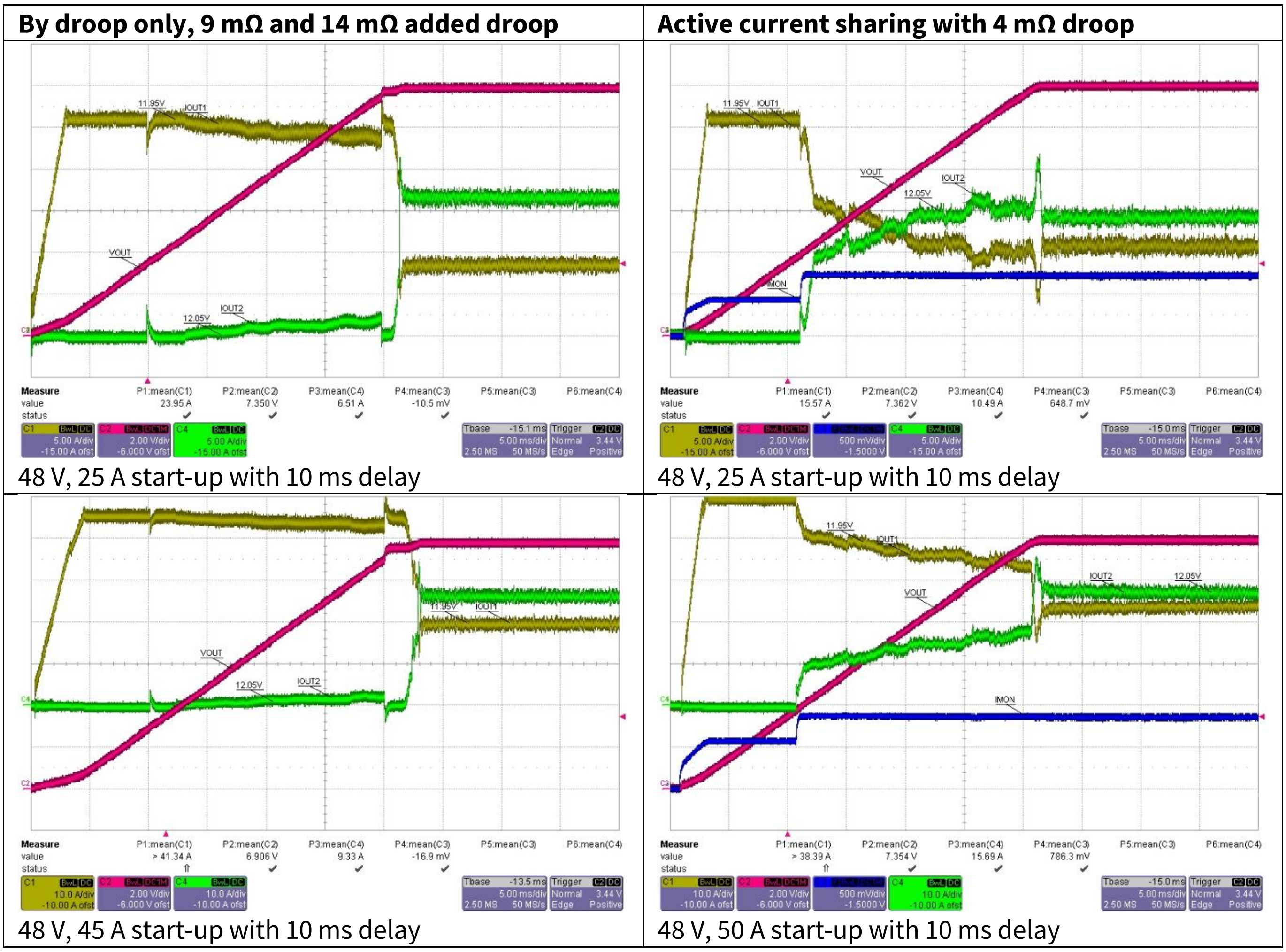
Test case 1: active current sharing is disabled, current sharing by droop only:

- Added droop is 14 mΩ at ramp and reduced to 9 mΩ at regulation
- vc0_vavp_kfp = 24
- vc0_vavp_clamp_neg = 0, vc0_vavp_clamp_pos = 24
- Enable DE start-up of unit #2

Test case 2: active current sharing is enabled:

- No added droop at ramp, standard droop 4 mΩ at regulation
- ishr_scale = 10, ishr_ki = 8, ishr_kp = 0
- ishare_clamp_neg = 2 (25 mV at V_{OUT}), ishare_clamp_pos = 10 (125 mV at V_{OUT})
- MFR_ISHARE_THRESHOLD = 0x0002 (2 A)
- Enable DE start-up of unit #2

Ch1: IOUT1 of unit#1, Ch2: V_{OUT} (2 V/div), Ch3: IMON pin (0.5 V/div), Ch4: IOUT2 of unit#2



By droop only, 9 mΩ and 14 mΩ added droop

Active current sharing with 4 mΩ droop

48 V, 25 A start-up with 10 ms delay

48 V, 25 A start-up with 10 ms delay

48 V, 45 A start-up with 10 ms delay

48 V, 50 A start-up with 10 ms delay

Figure 89 Start-up waveforms with different TON_DELAY

Please note: In this test, the IMON resistor is always connected in circuit (not controlled by PWM11).

5.7.4 Active current sharing under load transient

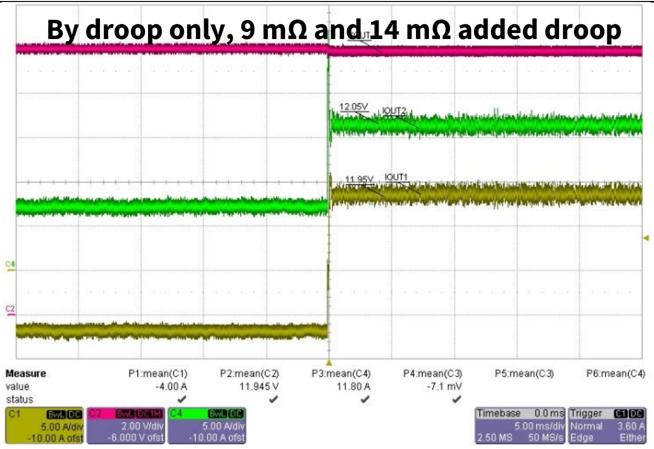
This section shows active current sharing during load transient. The configuration is the same as in test 5.7.2.

Unit #1, VOUT_COMMAND = 11.95 V, xaddr 0x41

Unit #2, VOUT_COMMAND = 12.05 V, xaddr 0x40

Ch1: IOUT1 of unit#1 (5 A/div), Ch2: V_{OUT} (2 V/div), Ch3: IMON pin (0.5 V/div), Ch4: IOUT2 of unit#2 (5 A/div)

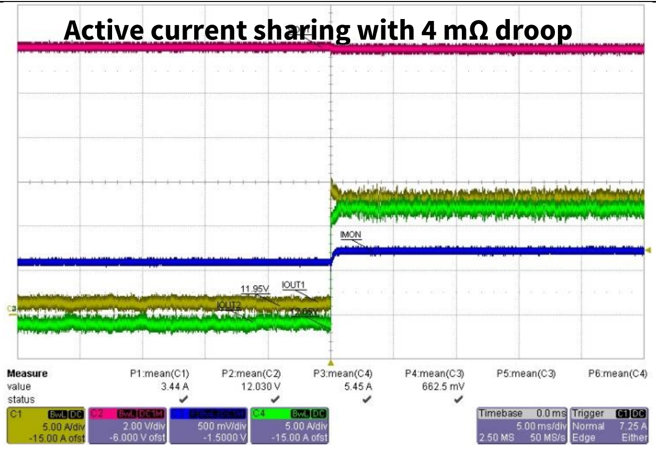
By droop only, 9 mΩ and 14 mΩ added droop



Measure	P1:mean(C1)	P2:mean(C2)	P3:mean(C4)	P4:mean(C3)	P5:mean(C3)	P6:mean(C4)
value	-4.00 A	11.945 V	11.80 A	-7.1 mV		
status	✓	✓	✓	✓		

48 V, 0 A to 25 A

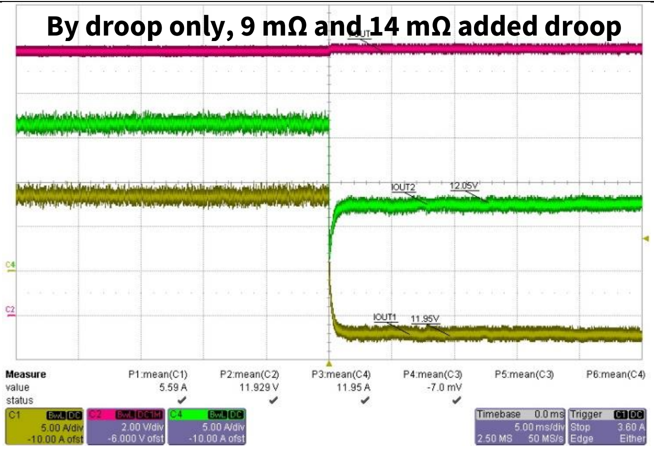
Active current sharing with 4 mΩ droop



Measure	P1:mean(C1)	P2:mean(C2)	P3:mean(C4)	P4:mean(C3)	P5:mean(C3)	P6:mean(C4)
value	3.44 A	12.030 V	5.45 A	662.5 mV		
status	✓	✓	✓	✓		

48 V, 0 A to 25 A

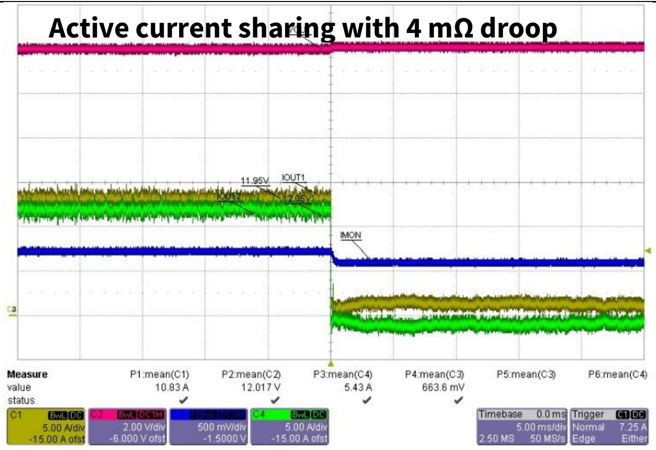
By droop only, 9 mΩ and 14 mΩ added droop



Measure	P1:mean(C1)	P2:mean(C2)	P3:mean(C4)	P4:mean(C3)	P5:mean(C3)	P6:mean(C4)
value	5.59 A	11.929 V	11.95 A	-7.0 mV		
status	✓	✓	✓	✓		

48 V, 25 A to 0 A

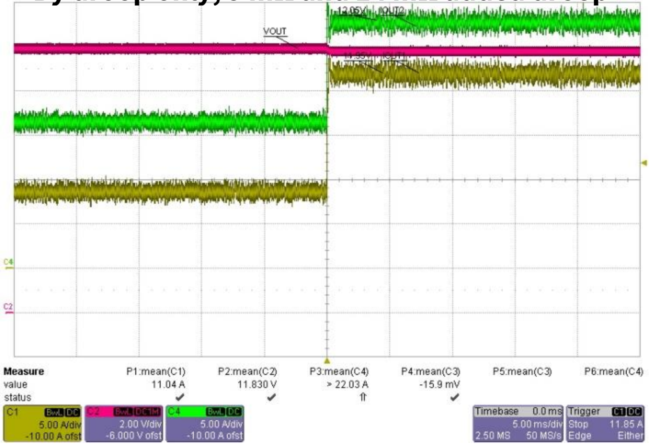
Active current sharing with 4 mΩ droop



Measure	P1:mean(C1)	P2:mean(C2)	P3:mean(C4)	P4:mean(C3)	P5:mean(C3)	P6:mean(C4)
value	10.83 A	12.017 V	5.43 A	663.6 mV		
status	✓	✓	✓	✓		

48 V, 25 A to 0 A

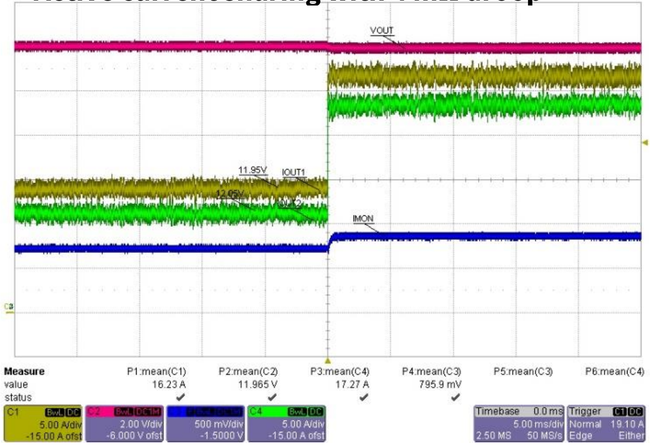
By droop only, 9 mΩ and 14 mΩ added droop



Measure	P1:mean(C1)	P2:mean(C2)	P3:mean(C4)	P4:mean(C3)	P5:mean(C3)	P6:mean(C4)
value	11.04 A	11.830 V	> 22.03 A	-15.9 mV		
status	✓	✓	⇑	✓		

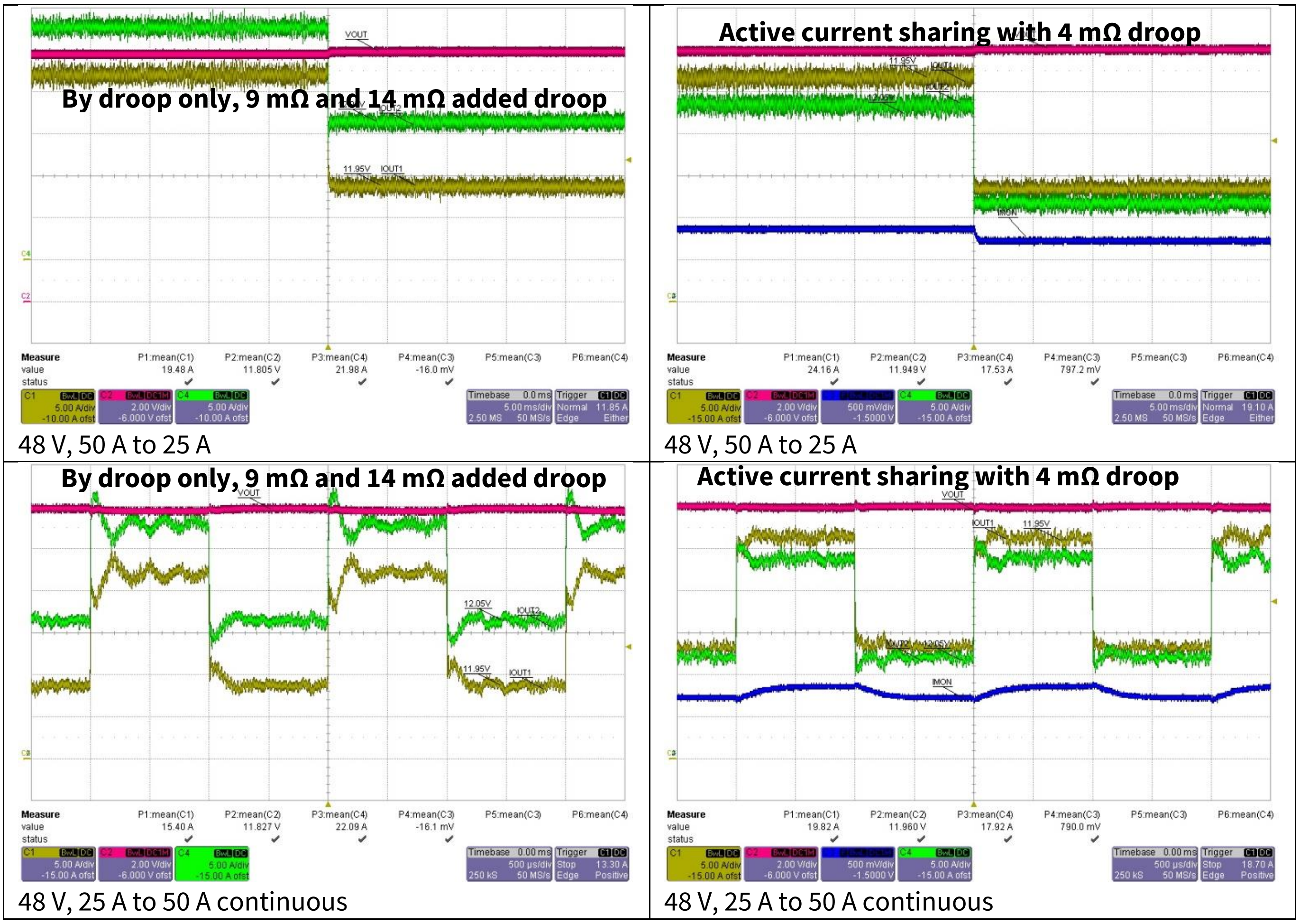
48 V, 25 A to 50 A

Active current sharing with 4 mΩ droop



Measure	P1:mean(C1)	P2:mean(C2)	P3:mean(C4)	P4:mean(C3)	P5:mean(C3)	P6:mean(C4)
value	16.23 A	11.965 V	17.27 A	795.9 mV		
status	✓	✓	✓	✓		

48 V, 25 A to 50 A



48 V, 50 A to 25 A

48 V, 50 A to 25 A

48 V, 25 A to 50 A continuous

48 V, 25 A to 50 A continuous

Figure 90 Current sharing at dynamic load

5.8 Summary of current sharing

This chapter describes the features of passive and active current sharing of the XDPP1100, and how to configure these features. Test results of the droop-based current sharing are compared with active current sharing. The active current sharing shows a better balance between two units when the output voltage has 100 mV mis-match. Start-up in pre-bias condition is also discussed.

6 Flux balancing

In the FB converter (**Figure 91**), PCMC is often used to balance the current in each half-cycle and to prevent transformer saturation. However, PCMC requires an additional current transformer to sense the primary current, and often loses accuracy during start-up when the duty-cycle is narrow. In telecom-brick 48 V to 12 V converters, it is common to run the converter in an open-loop condition at 36 V input to optimize transformer design. In this case, the converter switches at the maximum duty-cycle, which is very close to 100 percent. The current transformer may not have enough time to reset, and this could cause transformer saturation.

VMC doesn’t require a current transformer. On top of saving on BOM cost and PCB area, it also improves reliability. If flux balance can be achieved without additional cost, VMC is preferred. To avoid “flux walkaway” in VMC, standard techniques rely on some combination of over-sizing the transformer by using a larger core, placing a capacitor in series with the primary winding, or gapping the transformer core, which leads to increased core and conduction losses.

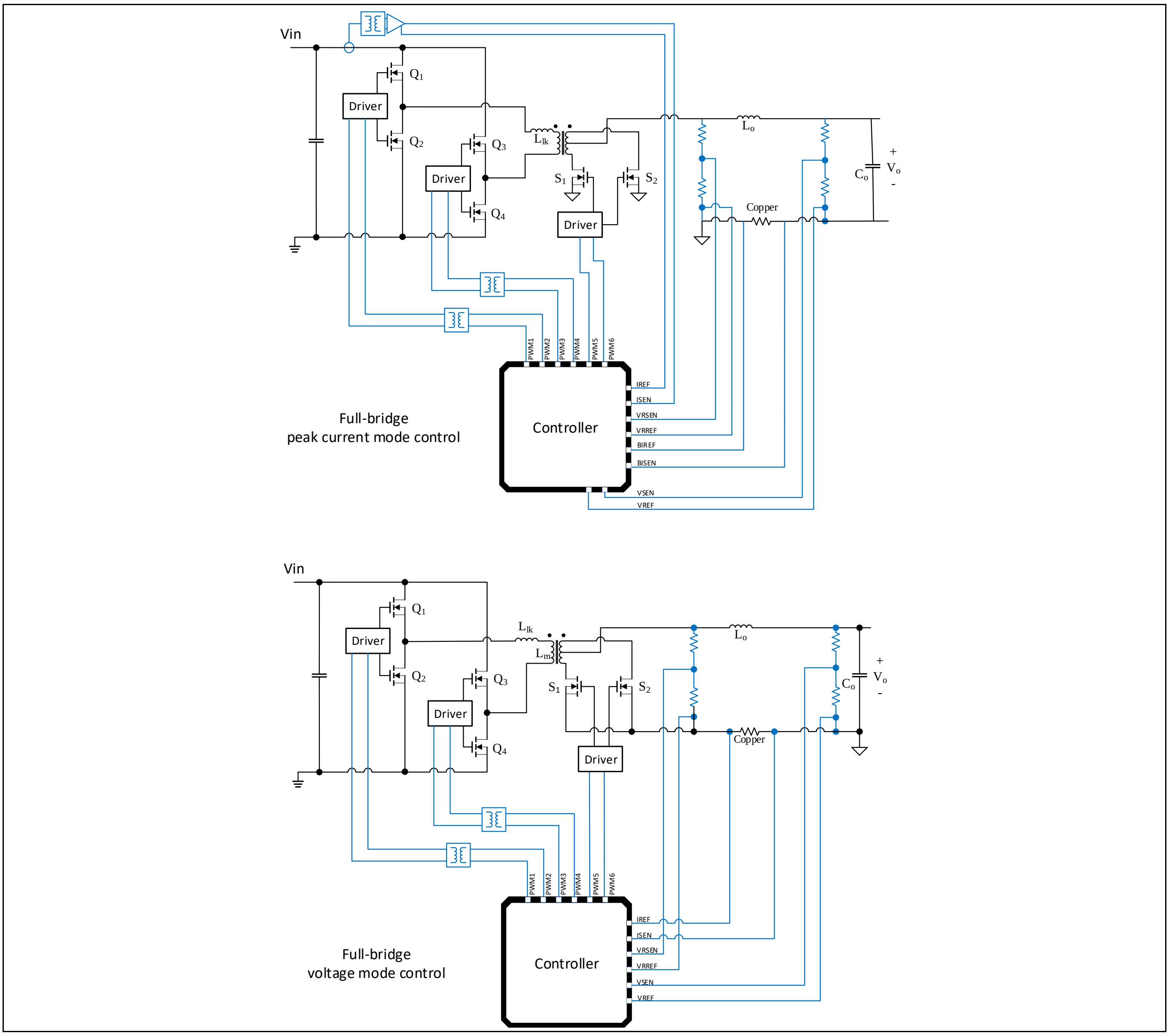


Figure 91 FB converter

Flux balancing

To enable use of the minimum core size, and to remove the bulky DC blocking capacitor, the transformer flux must be maintained within given bounds, or must be balanced by adjusting the PWM of each half-cycle to account for practical timing differences.

The XDPP1100 implements flux balancing by maintaining volt-second balance in each half-cycle. The voltage and timing are measured from the transformer secondary winding. The error between the volt-second products of each half-cycle is fed to a PI compensation network for duty-cycle adjustment.

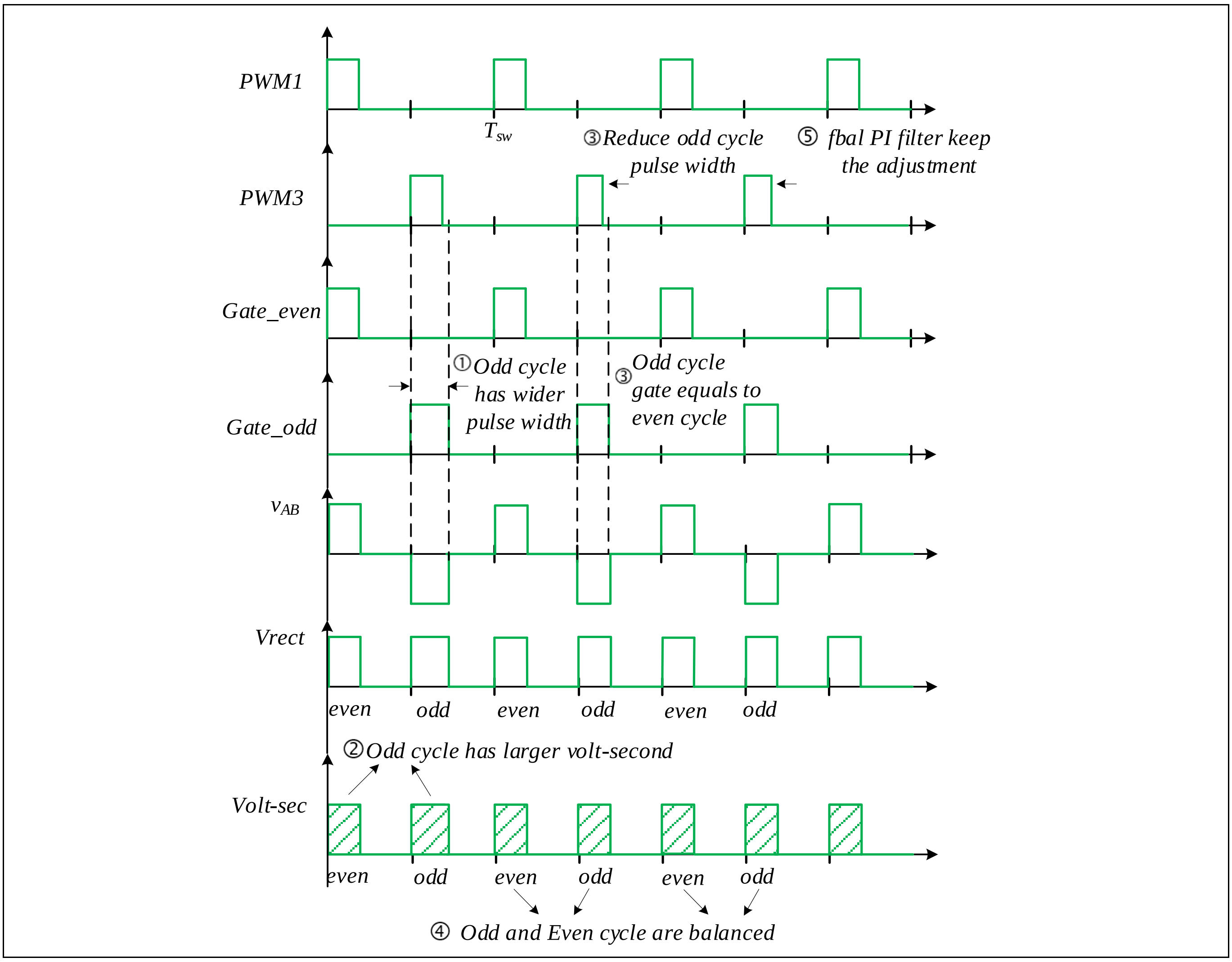

PWM1
T_{sw}
③Reduce odd cycle pulse width
⑤ fbal PI filter keep the adjustment
PWM3
Gate_even
①Odd cycle has wider pulse width
③Odd cycle gate equals to even cycle
Gate_odd
v_{AB}
Vrect
even
odd
even
odd
even
odd
②Odd cycle has larger volt-second
Volt-sec
even
odd
even
odd
even
odd
④ Odd and Even cycle are balanced

Figure 92 Volt-second balance

As shown in **Figure 92**, the volt-second flux balance can be described as follows:

- In this example, the even and odd half-cycles have different pulse widths due to gate driver delays. The odd cycle has a wider pulse.
- The V_{RECT} voltage follows the gate driver pulse width and is sensed by VRSEN. The odd cycle has a larger volt-second product.
- The flux balance block computes a duty-cycle adjustment and reduces the pulse width of the odd cycle. With the same gate driver delays, odd cycle gate pulse now is equal to even cycle.
- With the adjustment, the volt-second of the even and odd cycles is balanced.
- The flux balance block has a PI filter. The i-term of the PI filter maintains the duty-cycle adjustment in the presence of zero errors and maintains flux balance.

In the example in **Figure 92**, the even cycle is assigned to PWM1 and the odd cycle is assigned to PWM3. This mapping is not fixed but user configurable. The user could use the XDPP1100 GUI to select PWM mapping as per actual HW design.

To achieve high-performance flux balancing, accurate voltage and timing measurements are critical, especially timing measurement. This is because the voltage unbalancing is typically introduced by the mis-match of MOSFET on-state resistance ($R_{DS(on)}$) and the mis-match of PCB trace impedance of the power stage. In mid-voltage and HV applications, such as telecom brick converters or AC-DC server power supplies, the voltage error generated by $R_{DS(on)}$ is a tiny proportion of the input voltage. In addition, $R_{DS(on)}$ has a positive temperature coefficient. The mis-match due to $R_{DS(on)}$ could be self-corrected in steady-state. The branch with lower resistance has higher magnetizing current. It causes higher power loss in MOSFETs. $R_{DS(on)}$ would increase with junction temperature and offsets the initial $R_{DS(on)}$ mis-match.

Compared to voltage error, timing mis-match introduces a higher volt-second error in terms of percentage of the total. Mis-matched timing comes from mis-matched turn-on/turn-off time of the controller, gate driver delays, and insufficient PWM resolution. The XDPP1100 can lock the duty-cycle of the odd cycle to the even cycle, with PWM resolution of 78.125 ps. This eliminates two important sources of a timing mis-match. For gate driver delays, many gate drivers have very good delay matching between their internal channels, but the variation from part to part could be in the range of tens of nanoseconds. In the FB topology, the primary MOSFETs are often driven by two HB drivers. The mis-matched propagation delay between the two drivers dominates the timing mis-match. In high switching frequency applications, it becomes the major source of flux imbalance.

Many advanced digital controllers support V_{RECT} sensing from the transformer secondary side. However, sensing both voltage and pulse width (timing) is a challenge. The voltage sense ADC must have wide input voltage range, while retaining high resolution over the dynamic range. It should also be capable of responding to a fast-changing edge for proper timing measurement while simultaneously being immune to voltage glitches. The XDPP1100 offers the following benefits on V_{RECT} sensing:

- Fast V_{RECT} sensing over a large dynamic range (100 Msps, 0.5 V to 2.1 V)
- High-speed edge detector (200 MHz)
- Excellent noise immunity
- Programmable leading-edge blanking time
- Hardware-based implementation

Other challenges require attention in system-level design, i.e. good layout to avoid too much noise on the V_{RECT} waveform; proper snubber for a clean V_{RECT} waveform; avoiding a large filter to obtain a sharp V_{RECT} pulse shape.

When the XDPP1100 VADC is configured to measure the rectified voltage waveform (V_{RECT}), it enables the rectified voltage sense processor and VRS edge comparator (**Figure 93**). After the blanking window, sampling of the rectified voltage can occur (shown as the T_{sample} waveform). The sampling window ends when the associated PWM signal goes low. If the input voltage changes during this period, the ADC tracks the change. At the end of the sampling window, the ADC remembers the value of the last ADC sample and uses this value for the volt-second computation.

The edge comparator detects the rising edge and falling edge of the V_{RECT} waveform. The vsp counter starts counting the pulse width when the rising edge is detected and stops counting when the falling edge is detected. The counter resolution is 5 ns. Any filter added to the VRSEN pin would distort the VRS edge, leading to timing measurement errors. Thus, it is not recommended to add a filter cap to the VRSEN or BVRSEN pin when it is used for the flux balancing function.

More details on V_{RECT} sensing can be found in chapter 2.3.

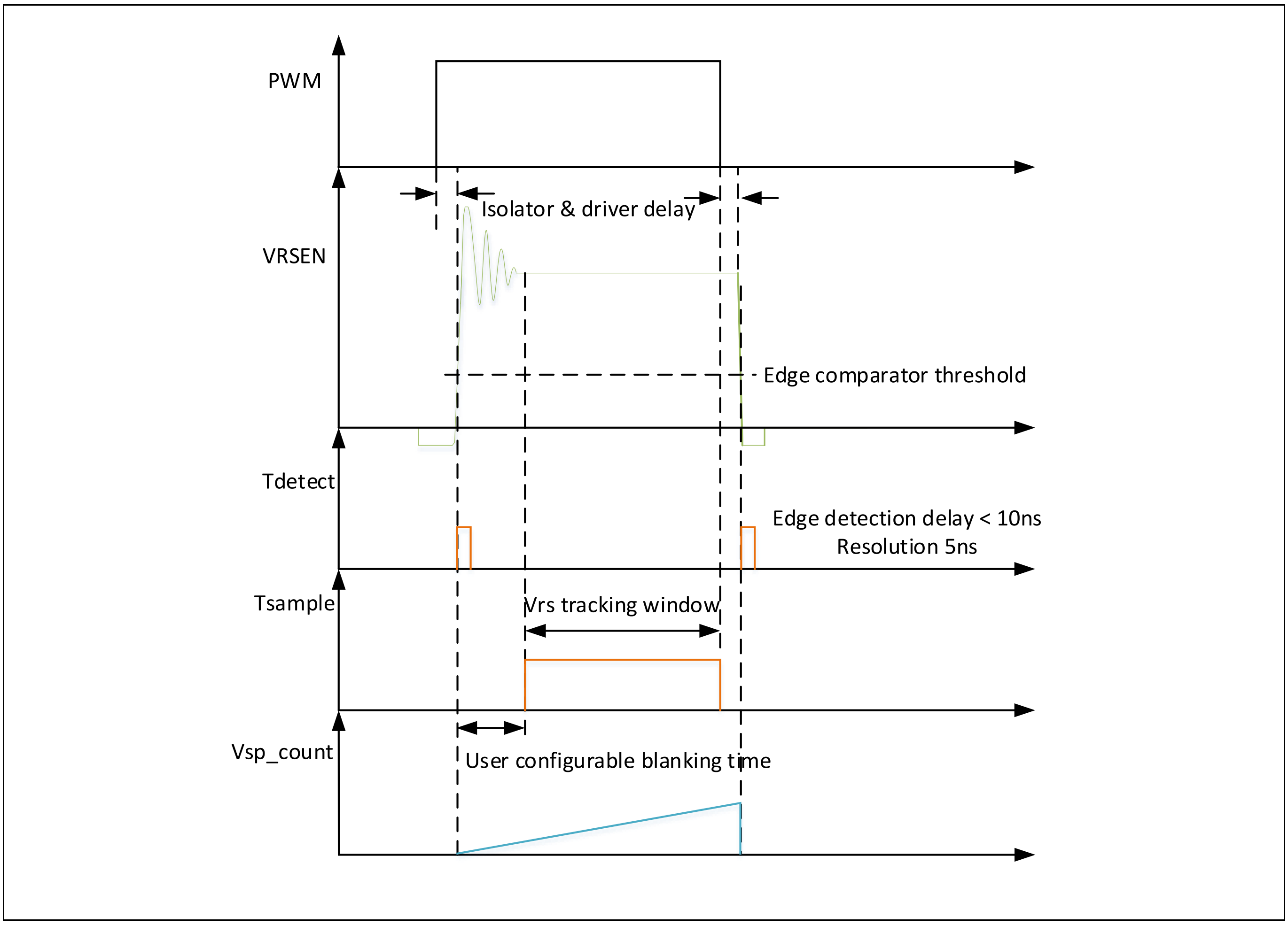


Figure 93 V_{RECT} sensing by VRSEN

6.1 XDPP1100 volt-second flux balance

The XDPP1100 has two flux balance blocks, both associated with loop 0. “fbal” is used for single-phase topologies or phase 1 of an interleaved topology. “fbal2” is used for the second phase of an interleaved bridge application. Note that the XDPP1100 is unable to balance flux on loop 1 in a dual-loop scenario due to the limited number of voltage sensors.

Flux balancing is automatically enabled by FW under the following conditions.

Enable “fbal” (**en_fbal** = 1):

- Topology = FB or HB
- Control mode = VMC
- Operation is enabled

Enable “fbal2” (**en_fbal2** = 1):

- Topology = FB or HB
- Control mode = VMC
- INTERLEAVE_ENABLE is true (INTERLEAVE_ENABLE is configured by PMBus command 0xC5 FW_CONFIG_REGULATION)
- Operation is enabled

The duty-cycle lock registers must be set to 1 for flux balance correction. Once the duty-cycle lock is enabled, the odd half-cycle duty-cycle is locked to the even half-cycle duty-cycle prior to applying any flux balance correction.

Table 41 Duty-cycle lock registers

Register name	Register value	Description
ramp0_dutyc_lock	1	Ramp 0 duty-cycle lock enabled (fbal)
ramp1_dutyc_lock	1	Ramp 1 duty-cycle lock enabled (fbal2)

6.1.1 Flux balancing

The XDPP1100 volt-second based flux balancing corrects duty-cycle based on the following equations. The volt_second_error is used to compute duty-cycle adjustment. The adjustment only applies to odd half-cycles.

$$volt\_second\_error = vdt \ + tdv \qquad (6.1)$$

$$vdt = (vrs\_vrect\_even + vrs\_vrect\_odd) \times (cnt\_vrscomp\_even - cnt\_vrscomp\_odd) \qquad (6.2)$$

$$tdv = (cnt\_vrscomp\_even + cnt\_vrscomp\_odd) \times (vrs\_vrect\_even - vrs\_vrect\_odd) \qquad (6.3)$$

The following registers are used for volt-second computation. Table 42 list the registers of vsp1 for fbal. A similar set of registers is available by vsp2 for fbal2.

Table 42 Status_Vsense registers for volt-second computation (vsp1)

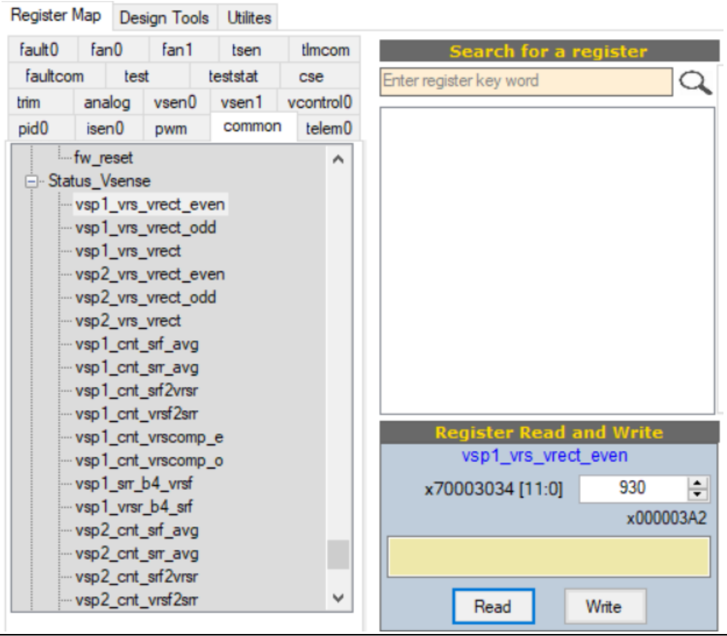
Register name	Description
vsp1_vrs_vrect_even	Measured VS1 (VRSEN) ADC rectification voltage on the even half-cycle. LSB = 1.25 mV, range = 0 to 5.11875 V
vsp1_vrs_vrect_odd	Measured VS1 (VRSEN) ADC rectification voltage on the odd half-cycle. LSB = 1.25 mV, range = 0 to 5.11875 V
vsp1_cnt_vrscomp_e	Non-averaged VRS1 VRS comp. pulse width measurement result for the even half-cycle of bridge topologies. LSB = 5 ns, range = 0 to 10235 ns
vsp1_cnt_vrscomp_o	Non-averaged VRS1 VRS comp. pulse width measurement result for the odd half-cycle of bridge topologies. LSB = 5 ns, range = 0 to 10235 ns

The status registers can be read from the “common” register tab in the XDPP1100 GUI, under the Status_Vsense folder. **vrs_vrect_even** and **vrs_vrect_odd** are the VRS ADC measured voltage after a LPF. The value of vrs_vrect can be estimated by input voltage V_{IN}, MFR_TRANSFORMER_SCALE, and the V_{RECT} resistor divider scale MFR_VRECT_SCALE.

MFR_TRANSFORMER_SCALE defines the transformer turns ratio; set this scale per N_S/N_P.

MFR_VRECT_SCALE is the resistor divider ratio of V_{RECT}.

For example, at 48 V input, transformer scale N_S/N_P = 1:3, V_{RECT} scale = 0.073, the **vrs_vrect_even** and **vrs_vrect_odd** are expected to be approximately equal to $\frac{48V \times \frac{1}{3} \times 0.073}{1.25mV} = 934$.



vsp1_vrs_vrect_even

x70003034 [11:0] 930

x000003A2

Figure 94 V_{SENSE} status registers

To enable VRS sensing for flux balancing, the VS ADC must be set to VRS mode (Table 43).

Table 43 VS ADC mode select registers

Register name	Register value	Description
vsp1_vrs_sel	1	V_{RECT} sensing (VRS) mode (for fbal)
vsp2_vrs_sel	1	V_{RECT} sensing (VRS) mode (for fbal2)

6.1.2 Voltage mode balance and time-only balance

Figure 95 shows the XDPP1100 flux balance block diagram. The highlighted signals are accessible through the register map. The XDPP1100 allows the user to select balance mode per the following configuration. The "voltage balance" adjusts the odd duty-cycle only based on the voltage of the odd and even cycles. The "time only balance" will adjust the odd duty-cycle only based on the pulse width of the odd and even cycles. The "voltage balance" can be selected in HB applications. The "time only balance" is useful when the voltage sense is not accurate due to parasitic ringing appearing on the V_{RECT} waveform. It could ignore the voltage mis-match and only corrects the timing mis-match.

Table 44 Volt-second balance mode

Balance mode	Register name	Value
Voltage balance	vbal_mode_sel	1
Flux balance (volt-second balance)	vbal_mode_sel	0
	fbal_time_only	0
Time only balance	vbal_mode_sel	0
	fbal_time_only	1

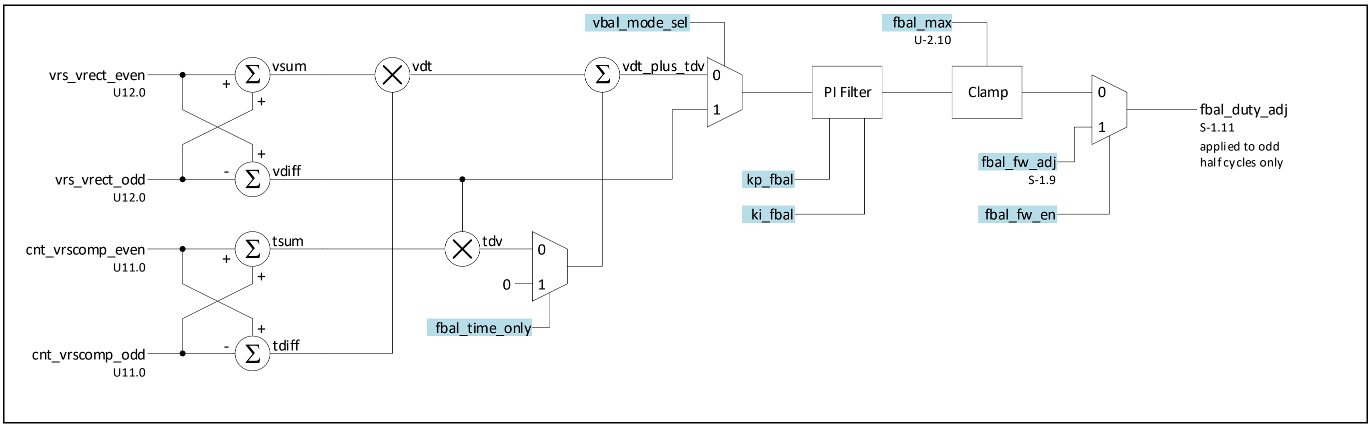


Figure 95 **XDPP1100 flux balancing block diagram**

6.1.3 Flux balancing PI filter

The PI filter consists of a proportional term that works on the instantaneous magnitude of the error, and an integral term that works on the magnitude and the duration of the error. The integral term is the sum of the instantaneous error over time, and it gives the accumulated error. The integral term sets how strongly the loop will response to the "past" information. The integral term set the low-frequency gain, and the proportional term sets the high-frequency gain. The magnitude response of the PI filter is defined by:

$$\sqrt{kp^2 + (\frac{ki}{2\pi \cdot Tsw \cdot f})^2} \tag{6.4}$$

Here, T_{SW} is the converter switching period.

Table 45 **Flux balancing PI filter register**

Register	Description	Equation

kp_fbal	Flux/voltage balancing PI filter proportional coefficient index	kp_exp = kp_fbal [5:3]
		kp_man = 8 + kp_fbal [2:0]
		kp = kp_man * 2^(kp_exp – 18)

ki_fbal	Flux/voltage balancing PI filter integral coefficient index	ki_exp = ki_fbal [5:3]
		ki_man = 8 + ki_fbal [2:0]
		ki = ki_man * 2^(ki_exp – 22)

Design example:

kp_fbal = 24_D = 011000_B, kp = 8 * 2^(3-18) = 0.000244

ki_fbal = 32_D = 100000_B, ki = 8 * 2^(4-22) = 3.05*10^-5

The PI filter magnitude over frequency can be plotted as **Figure 96**, at F_{sw} = 250 kHz.

Note: *Subscript #$_D$ means the data is in decimal format, #$_B$ means the data is in binary format.*

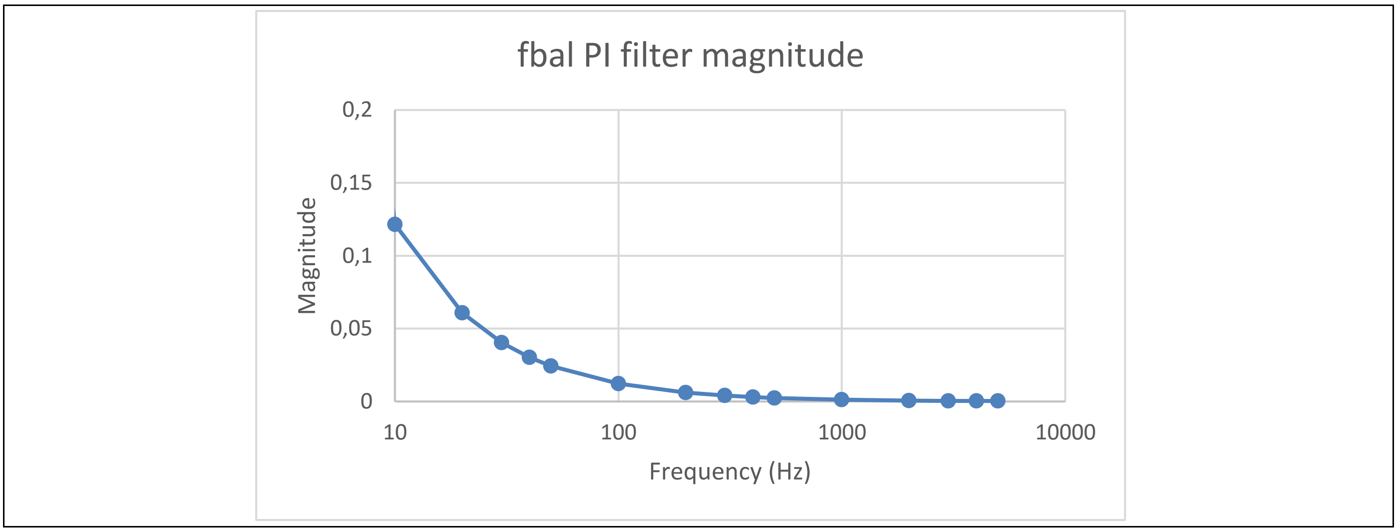


Figure 96 fbal PI filter example

6.1.4 Maximum limit of duty-cycle correction

The **fbal_max** register limits the maximum duty-cycle correction applied by the flux balance filter. The LSB of this register is 2^-10, and the range is from 0 to 24.902 percent.

For example, a FB converter has a switching frequency of 250 kHz. The estimated maximum timing mis-match is 40 ns. Expected maximum timing correction is 80 ns, which is 2 percent duty-cycle. Then **fbal_max** should be set to 20 for 2 percent maximum limit.

$$fbal\_max = \frac{2\%}{2^{-10}} = 20$$

Please note, setting **fbal_max** = 0 will block duty-cycle adjustment from the fbal function.

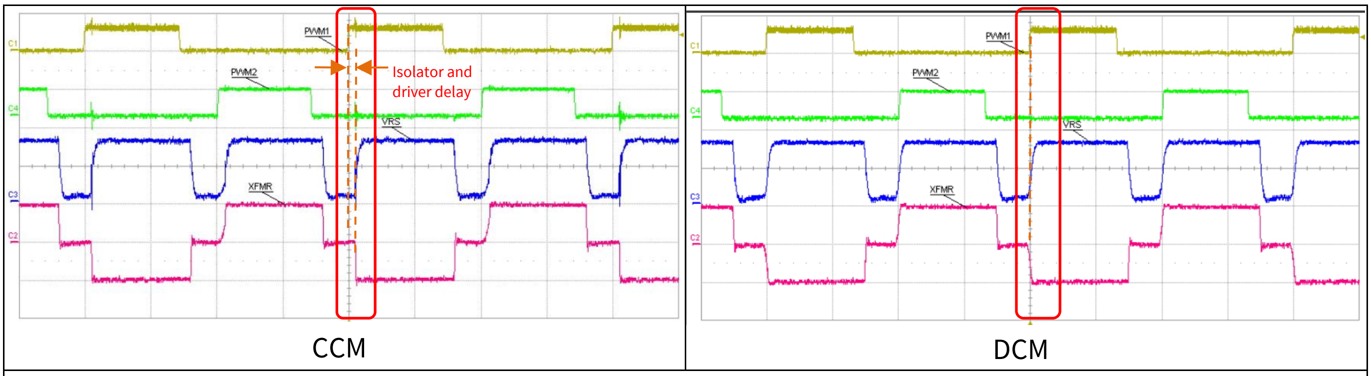
6.1.5 FW override

The XDPP1100 allows FW to override the flux balancing adjustment. When **fbalX_fw_en** (X = 1, 2) is high, the **fbalX_fw_adj** register overrides the computed flux balance duty-cycle adjustment. LSB of this register is 2^-9, and the range is from 0 to 24.805 percent.

6.1.6 DCM operation

At light load, output inductor current flows in a negative direction. If the secondary rectifier is not an SR MOSFET but a diode, the output current is discontinuous. This is termed discontinuous conduction mode (DCM) to indicate that the inductor current is discontinuous (diode mode) or negative (SR mode).

In DCM operation, the VRS rising edge happens before the primary PWM rising edge because the negative current in the inductor drives the VRS high as soon as the opposite SR turns off. That means the V_{RECT} pulse width cannot indicate primary PWM mis-match. Disabling flux balance in DCM is recommended. Flux balance DCM control registers are listed in Table 46.



Ch1: PWM output of even cycle, Ch4: PWM output of odd cycle,
Ch3: VRS sense voltage, Ch2: Transformer voltage

Figure 97 FB VRS voltage waveform in CCM and DCM

Table 46 Registers to disable flux balance in DCM

Register name	Description	Example
fbal_dcm_thresh	Determines where the fbal duty adjust will be removed in DCM Index of 63 disables feature LSB = 0.5 A, range = 0 to 31 A	Set to 20 for 10 A DCM threshold
fbal_dcm_dis_cnt	1 to 4, number of consecutive current samples below threshold to disable fbal	Set to 2, flux balance will be disabled after three consecutive current samples lower than fbal_dcm_thresh
fbal_dcm_ena_cnt	1 to 4, number of consecutive current samples above threshold to enable fbal	Set to 2, flux balance will be enabled after three consecutive current samples higher than fbal_dcm_thresh
fbal_dcm_0out_duty_adj	Determines whether in DCM to zero out fbal_duty_adj or freeze the current duty adjust	0 = freeze the current fbal_duty_adj 1 = zero out fbal_duty_adj Set to 1 is recommended

The DCM threshold varies with input voltage. Setting the **fbal_dcm_thresh** based on high-line operation could cover both low-line and high-line situations. A variable **fbal_dcm_thresh** per input voltage is also possible by FW patch.

Setting **fbal_dcm_dis_cnt**, **fbal_dcm_ena_cnt** to a higher number helps reduce jittering at the boundary of the threshold.

6.1.7 Flux balance fault protection

Failure to achieve flux balance within a programmable number of cycles should generate a fault. Flux balance fault is managed by a common fault block. **Figure 98** shows the flux balance fault block diagram. The highlighted signals are accessible through the register map.

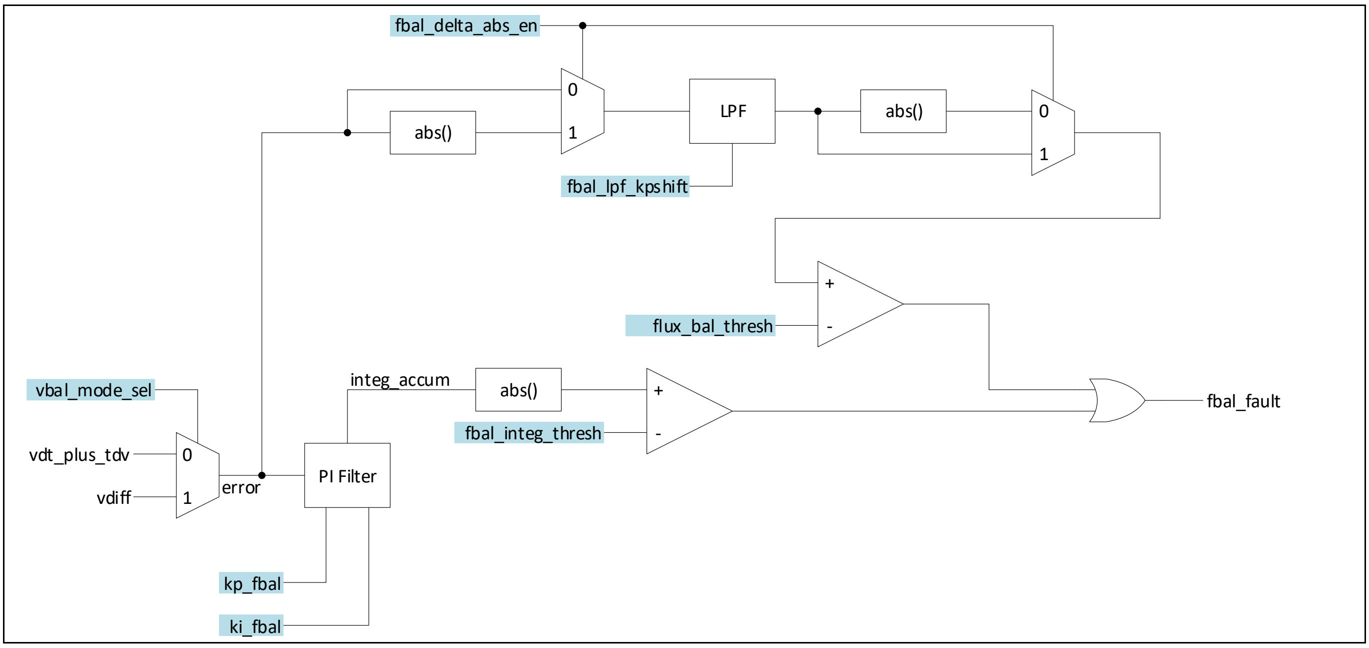


Figure 98 Flux balance fault protection

The flux balance fault can be detected in two ways: by absolute volt-second error, or by the accumulated integrator error.

The absolute flux balance error is calculated by the following equation:

$$fbal_{error} = [(vrs\_vrect\_even + vrs\_vrect\_odd) \times (cnt\_vrscomp\_even - cnt\_vrscomp\_odd) + (cnt\_vrscomp\_even + cnt\_vrscomp\_odd) \times (vrs\_vrect\_even - vrs\_vrect\_odd)]/256$$

The **flux_bal_thresh** defines flux balance fault threshold. LSB is 2. Set it to 0 to disable fbal error fault protection.

fbal_integ_thresh is the flux balance integrator error threshold defined with respect to maximum integrator range.

- 0 = disable
- 1 = 25 percent or greater (railed)
- 2 = 12.5 percent or greater
- 3 = 6.25 percent or greater

fbal_lpf_kpshift sets the LPF bandwidth at the output of fbal error. It bypasses the filter when set to 0. Set the filter to a lower bandwidth to reduce the sensitivity of fbal fault detection.

- Kp = 2^-Kpshift
- F3db = [kp/(1-kp)] * F_{switch} (kHz)/2 pi

For example, at 250 kHz switching frequency, **fbal_lpf_kpshift** = 1 sets LPF BW to 39.788 kHz. **fbal_lpf_kpshift** = 7 sets LPF BW to 0.313 kHz.

If the flux balance fault is tripped, the common fault will be reported by PMBus command STATUS_MFR_SPECIFIC (bit 4). The fbal fault is mapped to common fault bit [4], the fbal2 fault is mapped to common fault bit [7]. The common fault list is shown in Table 47. The common fault status can be checked by writing 23_D to PMBus command 0xFE MFR_FIRMWARE_COMMAND, and reading the result from PMBus command 0xFD MFR_FIRMWARE_COMMAND_DATA.

For example, after the fbal fault is triggered, write 23_D to PMBus command 0xFE and read 0xFD should return = 00 00 00 10. A common fault can be cleared by the CLEAR_FAULTS PMBus command.

The response to a flux balance fault is either to ignore or shut down. This is configured by the common fault shutdown mask register **fault_shut_mask_com**. Enabled faults disable both loop outputs and assert the shutdown interrupt. To enable fbal_fault shutdown, write 16_D to **fault_shut_mask_com**. To enable fbal2_fault shutdown, write 128_D to **fault_shut_mask_com**.

Table 47 Fault_shut_mask_com

Bit	Fault
0	Unused
1	Unused
2	IS1 (ISEN) tracking fault
3	IS2 (BISEN) tracking fault
4	fbal1_fault
5	IS1 (ISEN) PCL fault
6	IS1 (ISEN) SCP fault
7	fbal2_fault
8	IS2 (BISEN) PCL fault
9	IS2 (BISEN) SCP fault
10	Unused
11	VREF open fault
12	VSEN open fault
13	Unused
14	VRREF open fault
15	VRSEN open fault
16	Unused
17	BVREF_BVRREF open fault
18	BVSEN_BVRSEN open fault

Flux balancing is enabled at the beginning of the start-up ramp. The flux balance fault is blanked during the start-up ramp to prevent false triggering. The flux balance fault will be enabled when output voltage reaches the target.

6.2 Flux balance register descriptions

Table 48 describes the registers used by the flux balance function.

Table 48 Flux balance relevant register descriptions

Name	Address (loop 0/1)	Bits	Description
Common peripheral			
kp_fbal	7000_3000$_H$	[17:12]	Flux/voltage balance PI filter proportional coefficient index. Set to 0 to disable the proportional component of the filter. Note that index settings greater than 55 are clamped to 55. Note also that flux balancing requires that duty-cycle locking is enabled by rampX_dutyc_lock. • kp_exp = kp_fbal [5:3] • kp_man = 8 + kp_fbal [2:0] • kp = kp_man * 2^(kp_exp - 18)
ki_fbal	7000_3000$_H$	[23:18]	Flux/voltage balance PI filter integral coefficient index. Set to 0 to disable the integral component of the filter. Note that index settings greater than 55 are clamped to 55. Note also that flux balancing requires that duty-cycle locking is enabled by rampX_dutyc_lock. • ki_exp = ki_fbal [5:3] • ki_man = 8 + ki_fbal [2:0] • ki = ki_man * 2^(ki_exp - 22)
vbal_mode_sel	7000_3000$_H$	[24]	Flux/voltage balance filter mode select. 0 = flux balance mode 1 = voltage balance mode
fbal_time_only	7000_3000$_H$	[28]	In the flux balance mode of the flux/voltage balance PI filter, select between volt-second or time only balancing. 0 = volt-second balance mode 1 = time only balance mode
fbal_max	7000_3014$_H$	[30:23]	Flux/voltage balance maximum correction. This register limits the maximum duty-cycle correction applied by the flux balance filter. LSB of this register is 2^-10 Range is 0 to 24.902 percent Example: set fbal_max = 20 limits the maximum duty-cycle adjustment to 1.95 percent
vsp1_vrs_sel	7000_3018$_H$	[21]	VS1 (VRSEN) ADC VRS mode select. 0 = general-purpose ADC mode 1 = V_{RECT} sense (VRS) mode
vsp2_vrs_sel	7000_3018$_H$	[22]	VS2 (VRSEN) ADC rectification VRS mode select. 0 = V_{OUT} sense (VS) mode 1 = VRS mode
vrs_cmp_ref_sel	7000_3018$_H$	[27]	Rectification VRS comparator threshold select. This threshold is shared by VRS1 and VRS2. 0 = 500 mV 1 = 300 mV

Name	Address (loop 0/1)	Bits	Description
flux_bal_thresh	7000_3030$_H$	[7:0]	Flux balance fault threshold (0 = disable) LSB = 2 volt-seconds (µs) Equation used: [(ve+vo)(te-to)+(te+to)(ve-vo)]/256
fbal_integ_thresh	7000_3030$_H$	[9:8]	Flux balance integrator error threshold defined with respect to maximum integrator range. 0 = disable 1 = 25 percent or greater (railed) 2 = 12.5 percent or greater 3 = 6.25 percent or greater

fbal_lpf_kpshift	7000_3030$_H$	[12:10]	Coefficient of LPF at output of fbal error, set to 0 to bypass.
			Kp = 2^-Kpshift
			F3db = [kp/(1-kp)] * 50 MHz/2 pi
			Range 62.6 kHz to 7.95 MHz

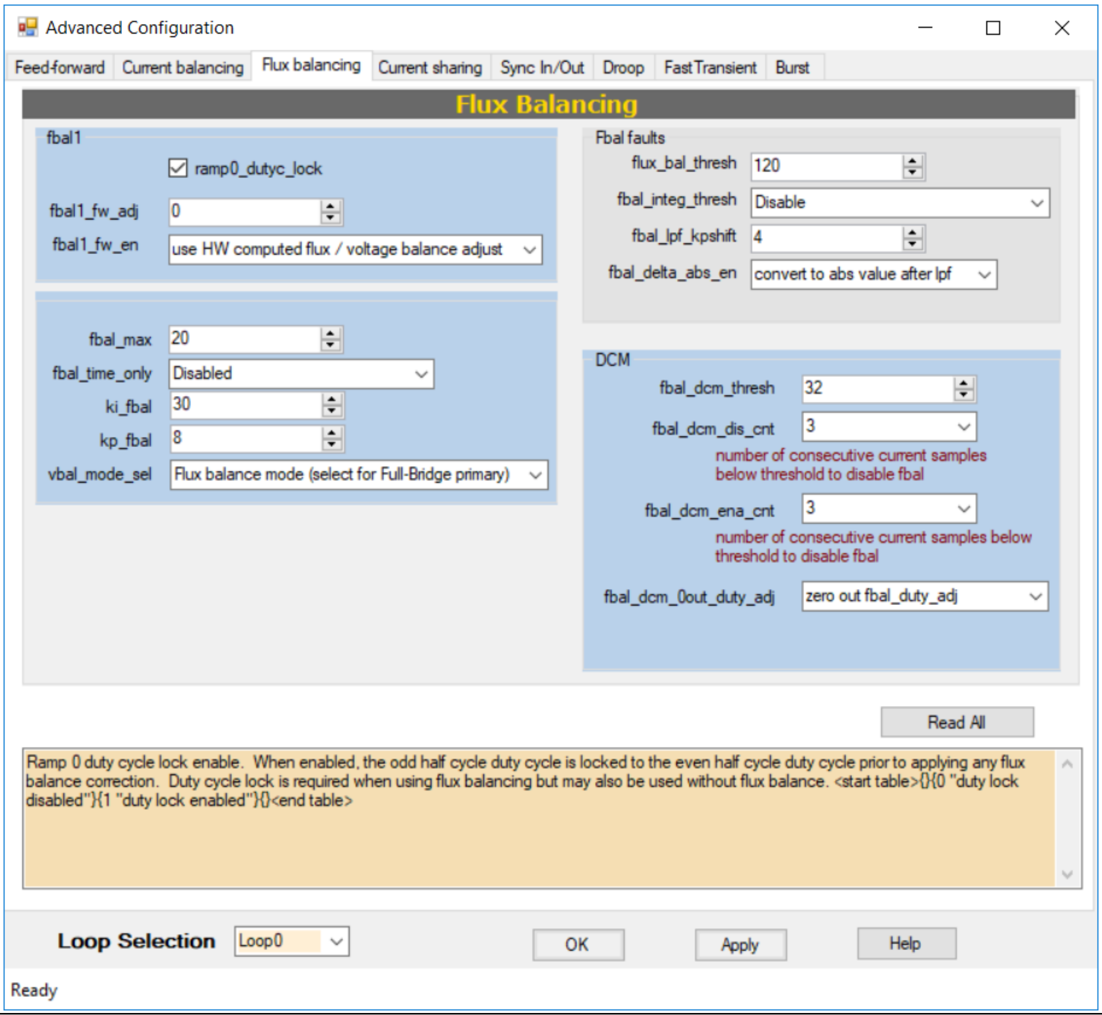
fbal_delta_abs_en	7000_3030$_H$	[13]	Determines whether absolute values to be applied to the error input before or after the LPF. 0 = convert to absolute value after LPF 1 = convert to abs value before LPF
fbal_dcm_thresh	7000_3030$_H$	[19:14]	Determines where the fbal duty adjust will be removed in DCM. Index of 63 disables feature LSB = 0.5 A, range = 0 to 31.5 A
fbal_dcm_dis_cnt	7000_3030$_H$	[21:20]	1 to 4, number of consecutive current samples below threshold to disable fbal. 0 = 1 sample 3 = 4 samples
fbal_dcm_ena_cnt	7000_3030$_H$	[23:22]	1 to 4, number of consecutive current samples above threshold to enable fbal. 0 = 1 sample 3 = 4 samples
fbal_dcm_0out_duty_adj	7000_3030$_H$	[24]	Determines whether in DCM to zero out fbal_duty_adj or freeze the current duty adjust. 0 = freeze the current fbal_duty_adj 1 = zero out fbal_duty_adj
fbal1_fw_adj	7000_307C$_H$	[7:0]	When fbal1_fw_en is high, this register overrides the loop 0, phase 1 HW flux/voltage balance duty-cycle adjusts output with a FW controlled setting. LSB of this register is 2^-9 Range is 0 to 24.805 percent
fbal1_fw_en	7000_307C$_H$	[8]	Enables FW controlled flux/voltage balance loop via fbal1_fw_adj.

Name	Address (loop 0/1)	Bits	Description
			0 = use HW computed flux/voltage balance adjust 1 = use fbal1_fw_adj
fbal2_fw_adj	7000_3080$_H$	[7:0]	When fbal2_fw_en is high, this register overrides the loop 0, phase 2 or loop 1 HW flux/voltage balance duty-cycle adjusts output with a FW controlled setting. LSB of this register is 2^-9 Range is 0 to 24.805 percent
fbal2_fw_en	7000_3080$_H$	[8]	Enables FW controlled flux/voltage balance loop via fbal2_fw_adj. 0 = use HW computed flux/voltage balance adjust 1 = use fbal2_fw_adj
vsp1_vrs_vrect_even	7000_3034$_H$	[11:0]	Measured VS1 (VRSEN) ADC rectification voltage on the even half-cycle. LSB = 1.25 mV, range = 0 to 5.11875 V
vsp1_vrs_vrect_odd	7000_3038$_H$	[11:0]	Measured VS1 (VRSEN) ADC rectification voltage on the odd half-cycle. LSB = 1.25 mV, range = 0 to 5.11875 V
vsp2_vrs_vrect_even	7000_3040$_H$	[11:0]	Measured VS2 (BVSEN_BVRSEN) ADC rectification voltage on the even half-cycle. LSB = 1.25 mV, range = 0 to 5.11875 V
vsp2_vrs_vrect_odd	7000_3044$_H$	[11:0]	Measured VS2 (BVSEN_BVRSEN) ADC rectification voltage on the odd half-cycle. LSB = 1.25 mV, range = 0 to 5.11875 V
vsp1_cnt_vrscomp_e	7000_3064$_H$	[10:0]	Non-averaged VRS1 VRS comp. pulse width measurement result for ACF topology or the even half-cycle of bridge topologies. LSB = 5 ns, range = 0 to 10235 ns
vsp1_cnt_vrscomp_o	7000_3064$_H$	[21:11]	Non-averaged VRS1 VRS comp. pulse width measurement result for ACF topology or the odd half-cycle of bridge topologies. LSB = 5 ns, range = 0 to 10235 ns
vsp2_cnt_vrscomp_e	7000_306C$_H$	[10:0]	Non-averaged VRS2 VRS comp. pulse width measurement result for ACF topology or the even half-cycle of bridge topologies. LSB = 5 ns, range = 0 to 10235 ns
vsp2_cnt_vrscomp_o	7000_306C$_H$	[21:11]	Non-averaged VRS2 VRS comp. pulse width measurement result for ACF topology or the odd half-cycle of bridge topologies. LSB = 5 ns, range = 0 to 10235 ns
PWM peripheral			
ramp0_dutyc_lock	7000_2C00$_H$	[16]	Ramp 0 duty-cycle lock enable. When enabled, the odd half-cycle duty-cycle is locked to the even half-cycle duty-cycle prior to applying any flux balance correction. Duty-

Name	Address (loop 0/1)	Bits	Description
			cycle lock is required when using flux balancing but may also be used without flux balance. 0 = duty lock disabled 1 = duty lock enabled
ramp1_dutyc_lock	7000_2C00$_H$	[17]	Ramp 1 duty-cycle lock enable. When enabled, the odd half-cycle duty-cycle is locked to the even half-cycle duty-cycle prior to applying any flux balance correction. Duty-cycle lock is required when using flux balancing but may also be used without flux balance. 0 = duty lock disabled 1 = duty lock enabled
Faultcom peripheral			
fault_shut_mask_com	7000_5400$_H$	[31:0]	Shutdown mask for “common” faults. Individual faults are enabled for shutdown when their corresponding bit is high. Enabled faults disable both loop outputs and assert the shutdown interrupt. The list shows the bit and its corresponding fault. 2: IS1 (ISEN) tracking fault 3: IS2 (BISEN) tracking fault 4: fbal1_fault 5: IS1 (ISEN) PCL fault 6: IS1 (ISEN) SCP fault 7: fbal2_fault 8: IS2 (BISEN) PCL fault 9: IS2 (BISEN) SCP fault 11: VREF open fault 12: VSEN open fault 14: VRREF open fault 15: VRSEN open fault 17: BVREF_BVRREF open fault 18: BVSEN_BVRSEN open fault

6.3 XDPP1100 GUI design tool for flux balance configuration

The XDPP1100 GUI design tool for flux balancing is shown in **Figure 99**.



fbal1

☑ ramp0_dutyc_lock

fbal1_fw_adj: 0

fbal1_fw_en: use HW computed flux / voltage balance adjust

fbal_max: 20

fbal_time_only: Disabled

ki_fbal: 30

kp_fbal: 8

vbal_mode_sel: Flux balance mode (select for Full-Bridge primary)

Fbal faults

flux_bal_thresh: 120

fbal_integ_thresh: Disable

fbal_lpf_kpshift: 4

fbal_delta_abs_en: convert to abs value after lpf

DCM

fbal_dcm_thresh: 32

fbal_dcm_dis_cnt: 3

number of consecutive current samples below threshold to disable fbal

fbal_dcm_ena_cnt: 3

number of consecutive current samples below threshold to disable fbal

fbal_dcm_0out_duty_adj: zero out fbal_duty_adj

Ramp 0 duty cycle lock enable. When enabled, the odd half cycle duty cycle is locked to the even half cycle duty cycle prior to applying any flux balance correction. Duty cycle lock is required when using flux balancing but may also be used without flux balance. <start table>{}{0 "duty lock disabled"}{1 "duty lock enabled"}{}<end table>

Loop Selection: Loop0

Figure 99 GUI design tools – flux balance

6.4 Test result of FB converter with flux balancing

In this section, start-up, load and line transient are tested with the XDPP1100 600 W 12 V/50 A FB evaluation board.

Figure 100 is the schematic of the power stage of the FB converter. The schematic of the control circuit is shown in **Figure 101**.

- XDPP1100 control board: R1, R2, C18, R15, R16, R45 are not used.

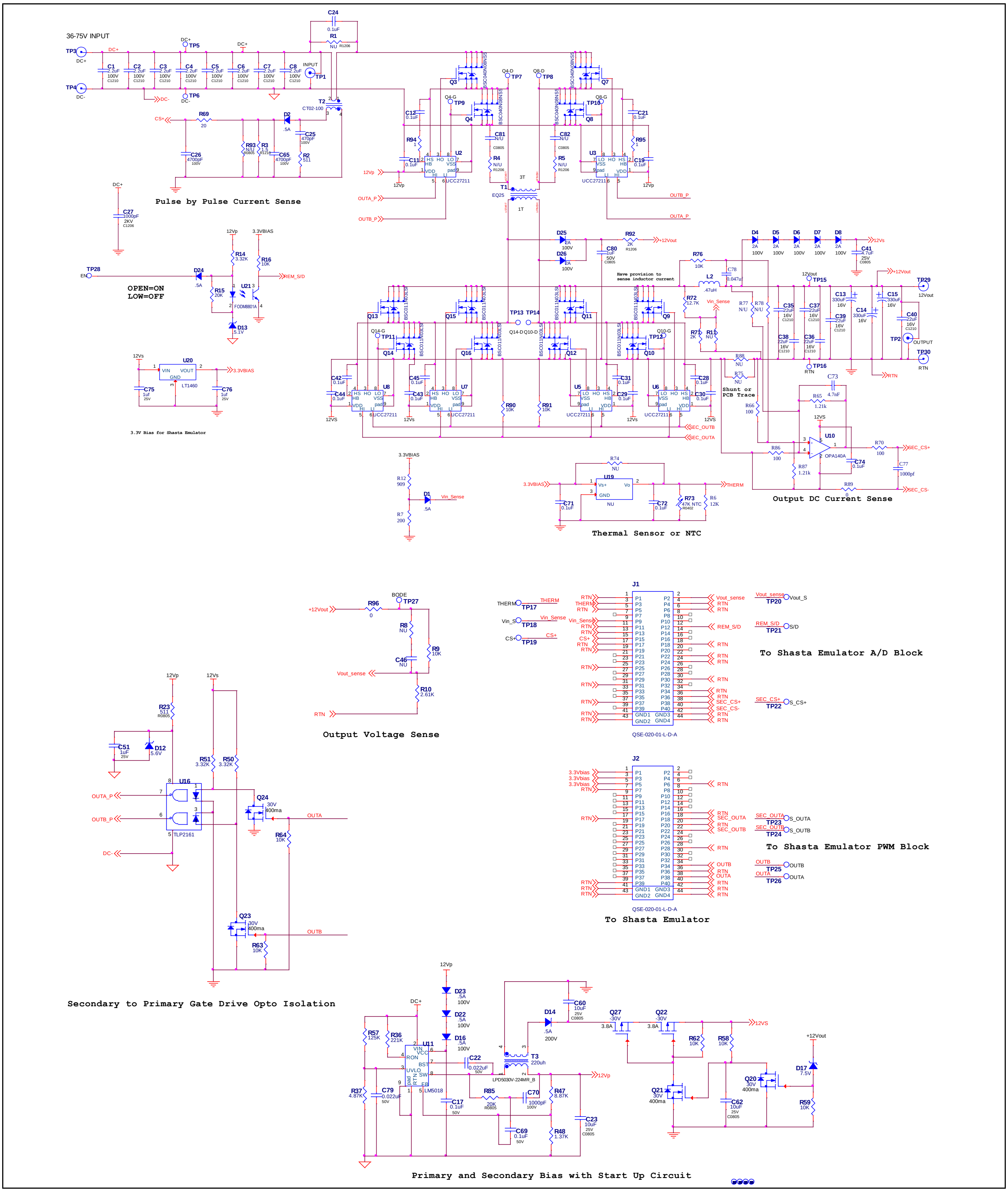

36-75V INPUT
Pulse by Pulse Current Sense
OPEN=ON
LOW=OFF
3.3V Bias for Shasta Emulator
Have provision to sense inductor current
Shunt or PCB Trace
Output DC Current Sense
Thermal Sensor or NTC
Output Voltage Sense
To Shasta Emulator A/D Block
QSE-020-01-L-D-A
To Shasta Emulator PWM Block
To Shasta Emulator
Secondary to Primary Gate Drive Opto Isolation
Primary and Secondary Bias with Start Up Circuit
UCC27211
BSC040N08NS5
BSC011N03LS
OPA140A
LT1460
TLP2161
FODM8801A
LM5018
LPD5030V-224MR_B
CT02-100
EQ25

Figure 100 Schematic of FB-FB – power stage

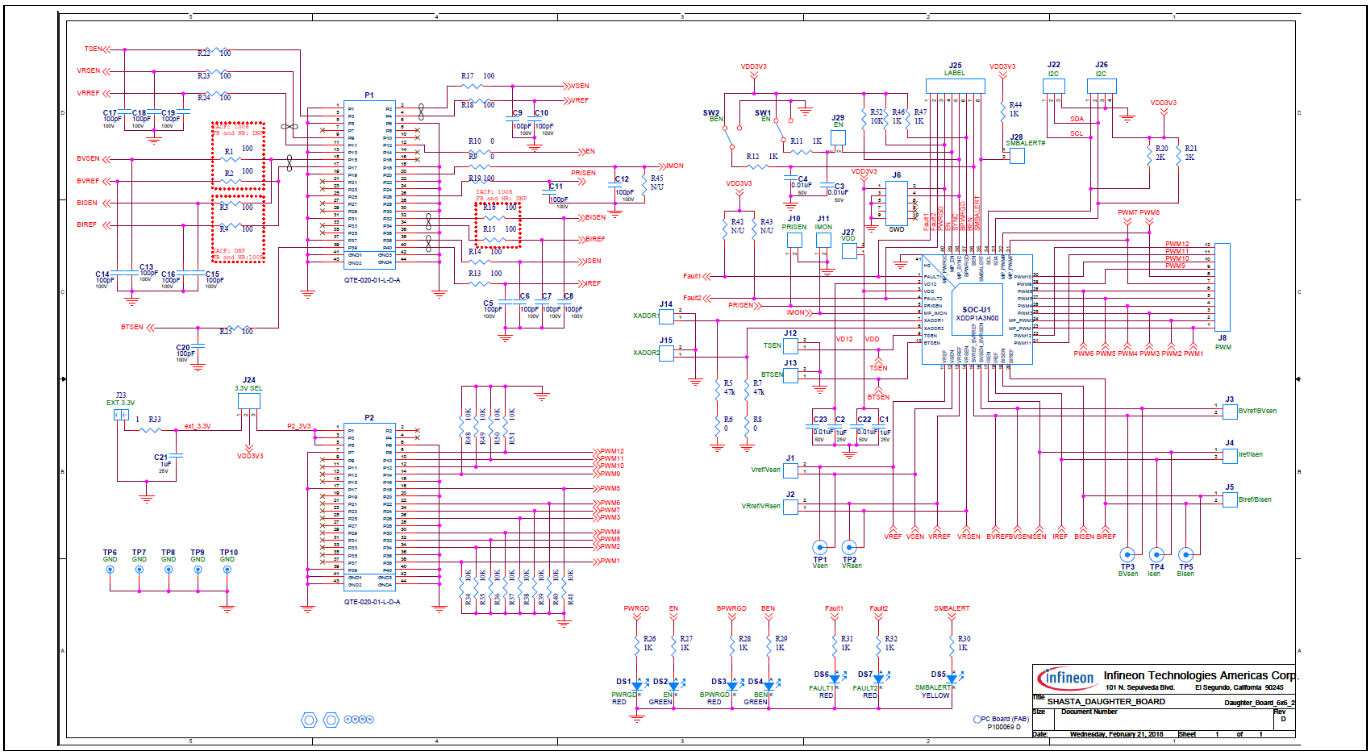


Figure 101 Schematic of the XDPP1100 control circuit

6.4.1 Flux balancing register configuration

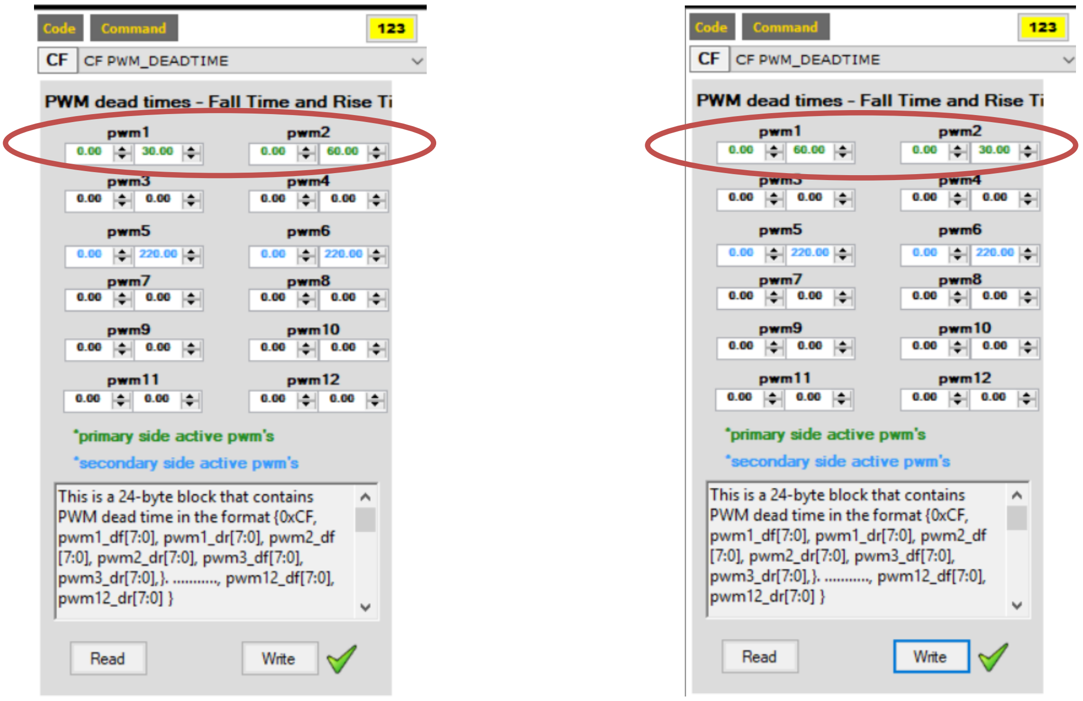
Table 49 is the configuration of FB-FB VMC with flux balancing.

Table 49 FB-FB VMC flux balancing configuration

Register name	Register value	Meaning
mode_control_loop0	0	Voltage mode control
ramp0_dutyc_lock	1	Ramp 0 duty-cycle lock enabled (fbal)
kp_fbal	8	
ki_fbal	30	
vbal_mode_sel	0	Flux/voltage balance filter mode select 0 = flux balance mode
fbal_time_only	0	Select between volt-second or time only balancing 0 = volt-second balance mode
fbal_max	20	Flux/voltage balance max. duty-cycle correction 1.95 percent
vsp1_vrs_sel	1	VS1 (VRSEN) ADC rectification voltage sense (VRS) mode select 1 = VRS mode
fbal1_fw_en	0	FW override is disabled
vsp1_vrs_cnt_num_avg	0	Time measurement averaged every four samples

6.4.2 Dead-time configuration for mis-matched pulse width

In this design, PWM1 and PWM2 are mapped to drive primary MOSFETs. PWM1 drives even cycles; PWM2 drives odd cycles. A 30 ns PW mis-match is forced to the primary gate drive by changing the dead-time configuration as shown in Figure 102 (a) and (b). The experiment shows how the XDPP1100 controller corrects the duty-cycle mis-match and flux imbalance. Please note that the optocoupler of the power board has 100 ns delay, thus the SR rise time is set to 220 ns.



CF — CF PWM_DEADTIME

PWM dead times - Fall Time and Rise Ti

pwm1	0.00	30.00	pwm2	0.00	60.00
pwm3	0.00	0.00	pwm4	0.00	0.00
pwm5	0.00	220.00	pwm6	0.00	220.00
pwm7	0.00	0.00	pwm8	0.00	0.00
pwm9	0.00	0.00	pwm10	0.00	0.00
pwm11	0.00	0.00	pwm12	0.00	0.00

*primary side active pwm's
*secondary side active pwm's

This is a 24-byte block that contains PWM dead time in the format {0xCF, pwm1_df[7:0], pwm1_dr[7:0], pwm2_df [7:0], pwm2_dr[7:0], pwm3_df[7:0], pwm3_dr[7:0],}. ...., pwm12_df[7:0], pwm12_dr[7:0] }

(a) PWM1 dead-time is 30 ns longer

CF — CF PWM_DEADTIME

PWM dead times - Fall Time and Rise Ti

pwm1	0.00	60.00	pwm2	0.00	30.00
pwm3	0.00	0.00	pwm4	0.00	0.00
pwm5	0.00	220.00	pwm6	0.00	220.00
pwm7	0.00	0.00	pwm8	0.00	0.00
pwm9	0.00	0.00	pwm10	0.00	0.00
pwm11	0.00	0.00	pwm12	0.00	0.00

*primary side active pwm's
*secondary side active pwm's

This is a 24-byte block that contains PWM dead time in the format {0xCF, pwm1_df[7:0], pwm1_dr[7:0], pwm2_df [7:0], pwm2_dr[7:0], pwm3_df[7:0], pwm3_dr[7:0],}. ...., pwm12_df[7:0], pwm12_dr[7:0] }

(b) PWM2 dead-time is 30 ns longer

Figure 102 Flux balancing PW mis-match by dead-time configuration

Figure 103 shows the waveform of configuration (a). PWM1 PW is 30 ns longer than PWM2 PW.

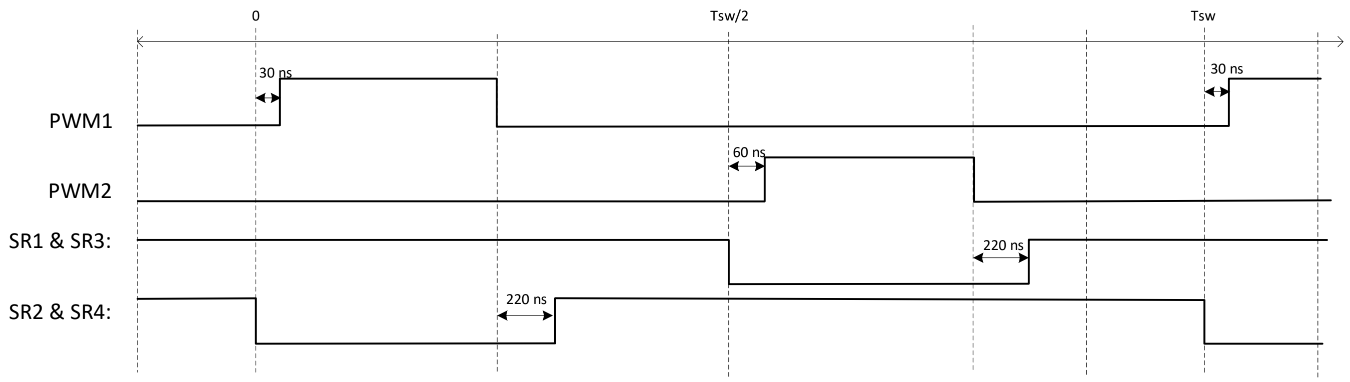


Figure 103 FB-FB gate drive waveform with mis-matched dead-time

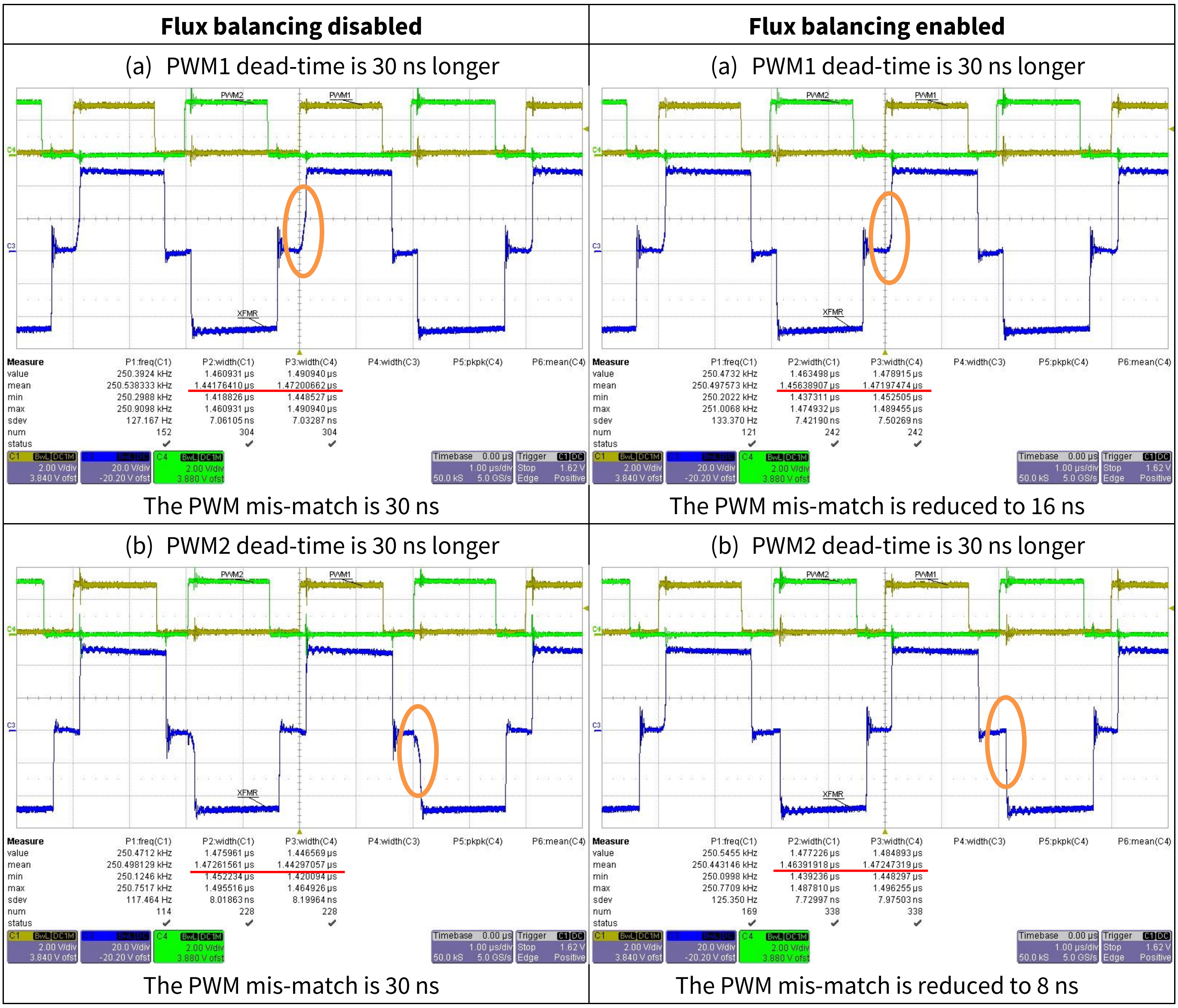
6.4.3 Steady-state waveform

Steady-state waveforms are tested at 48 V input, 20 A load. The converter operates in CCM. **Figure 104** shows the steady-state waveforms with flux balancing correction disabled (**fbal_max** = 0) and enabled (**fbal_max** = 20). The dead-time of primary PWM has 30 ns mis-match. The transformer waveform shows the mis-match of dead-time causing imbalance in the transformer, but flux balancing could correct the error in either direction.

Ch1: PWM1 output of even cycle

Ch4: PWM2 output of odd cycle

Ch3: Transformer primary voltage



Flux balancing disabled

(a) PWM1 dead-time is 30 ns longer

Measure	P1:freq(C1)	P2:width(C1)	P3:width(C4)	P4:width(C3)	P5:pkpk(C4)	P6:mean(C4)
value	250.3924 kHz	1.460931 µs	1.490940 µs			
mean	250.538333 kHz	1.44176410 µs	1.47200662 µs			
min	250.2988 kHz	1.418826 µs	1.448527 µs			
max	250.9098 kHz	1.460931 µs	1.490940 µs			
sdev	127.167 Hz	7.06105 ns	7.03287 ns			
num	152	304	304			
status	✓	✓	✓			

C1 BwL DC1M 2.00 V/div 3.840 V ofst
BwL DC 20.0 V/div -20.20 V ofst
C4 BwL DC1M 2.00 V/div 3.880 V ofst
Timebase 0.00 µs 1.00 µs/div 50.0 kS 5.0 GS/s
Trigger C1 DC Stop 1.62 V Edge Positive

The PWM mis-match is 30 ns

Flux balancing enabled

(a) PWM1 dead-time is 30 ns longer

Measure	P1:freq(C1)	P2:width(C1)	P3:width(C4)	P4:width(C3)	P5:pkpk(C4)	P6:mean(C4)
value	250.4732 kHz	1.463498 µs	1.478915 µs			
mean	250.497573 kHz	1.45638907 µs	1.47197474 µs			
min	250.2022 kHz	1.437311 µs	1.452505 µs			
max	251.0068 kHz	1.474932 µs	1.489455 µs			
sdev	133.370 Hz	7.42190 ns	7.50269 ns			
num	121	242	242			
status	✓	✓	✓			

C1 BwL DC1M 2.00 V/div 3.840 V ofst
BwL DC 20.0 V/div -20.20 V ofst
C4 BwL DC1M 2.00 V/div 3.880 V ofst
Timebase 0.00 µs 1.00 µs/div 50.0 kS 5.0 GS/s
Trigger C1 DC Stop 1.62 V Edge Positive

The PWM mis-match is reduced to 16 ns

(b) PWM2 dead-time is 30 ns longer

Measure	P1:freq(C1)	P2:width(C1)	P3:width(C4)	P4:width(C3)	P5:pkpk(C4)	P6:mean(C4)
value	250.4712 kHz	1.475961 µs	1.446569 µs			
mean	250.498129 kHz	1.47261561 µs	1.44297057 µs			
min	250.1246 kHz	1.452234 µs	1.420094 µs			
max	250.7517 kHz	1.495516 µs	1.464926 µs			
sdev	117.464 Hz	8.01863 ns	8.19964 ns			
num	114	228	228			
status	✓	✓	✓			

C1 BwL DC1M 2.00 V/div 3.840 V ofst
BwL DC 20.0 V/div -20.20 V ofst
C4 BwL DC1M 2.00 V/div 3.880 V ofst
Timebase 0.00 µs 1.00 µs/div 50.0 kS 5.0 GS/s
Trigger C1 DC Stop 1.62 V Edge Positive

The PWM mis-match is 30 ns

(b) PWM2 dead-time is 30 ns longer

Measure	P1:freq(C1)	P2:width(C1)	P3:width(C4)	P4:width(C3)	P5:pkpk(C4)	P6:mean(C4)
value	250.5455 kHz	1.477226 µs	1.484893 µs			
mean	250.443146 kHz	1.46391918 µs	1.47247319 µs			
min	250.0998 kHz	1.439236 µs	1.448297 µs			
max	250.7709 kHz	1.487810 µs	1.496255 µs			
sdev	125.350 Hz	7.72997 ns	7.97503 ns			
num	169	338	338			
status	✓	✓	✓			

C1 BwL DC1M 2.00 V/div 3.840 V ofst
BwL DC 20.0 V/div -20.20 V ofst
C4 BwL DC1M 2.00 V/div 3.880 V ofst
Timebase 0.00 µs 1.00 µs/div 50.0 kS 5.0 GS/s
Trigger C1 DC Stop 1.62 V Edge Positive

The PWM mis-match is reduced to 8 ns

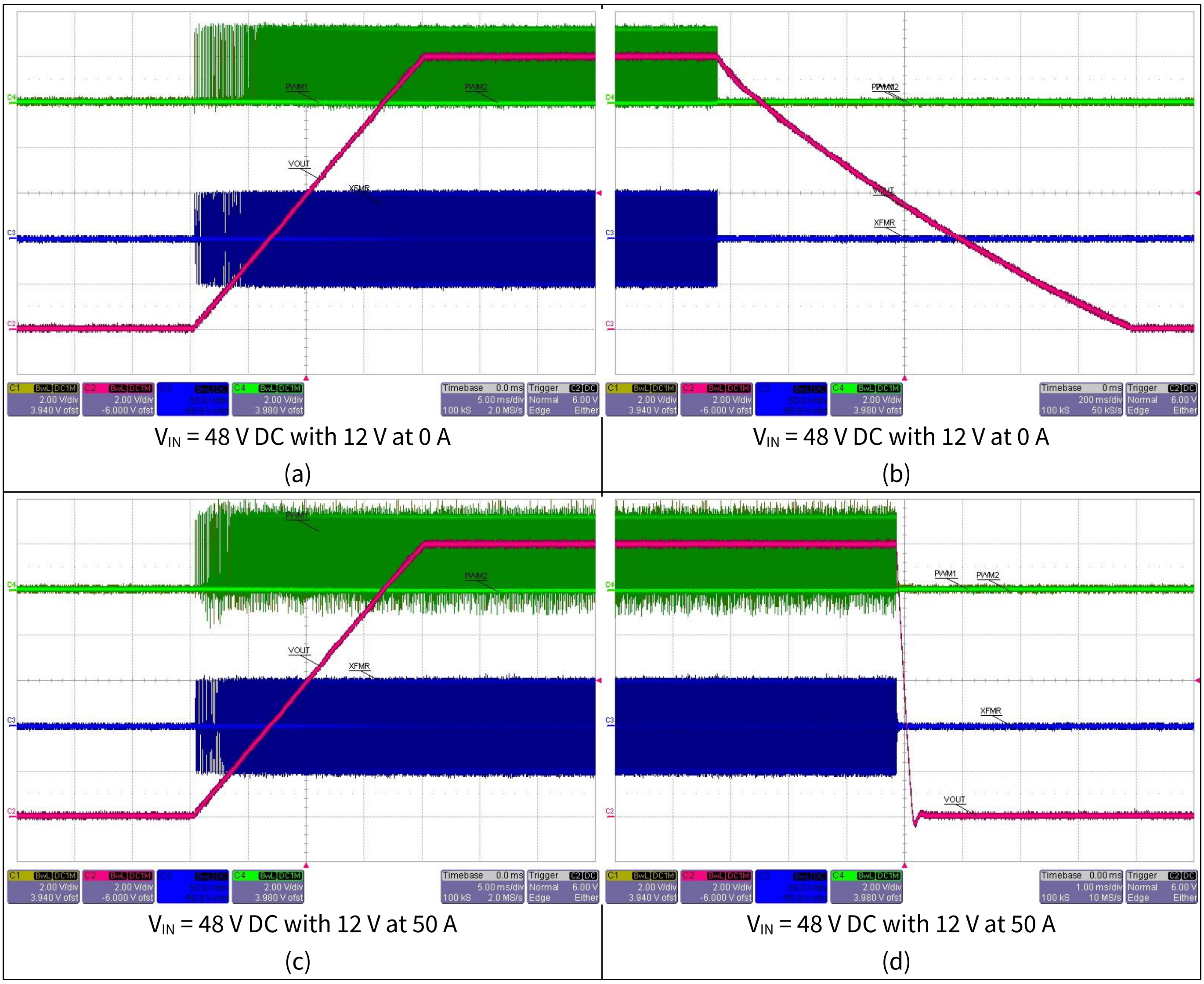
Figure 104 PWM and VRS waveforms after flux balance correction

6.4.4 Start-up and shutdown waveforms

This section shows switching waveforms during turn-on and turn-off with 30 ns primary gate drive timing imbalance. The “time only balance” is selected during the test.

Figure 105 shows the start-up and shutdown waveforms at 48 V input, and 12 V V_{OUT}.

Ch1 = primary drive PWM1, Ch2 = V_{OUT}, Ch3 = primary-side transformer winding, Ch4 = primary drive PWM2



V_{IN} = 48 V DC with 12 V at 0 A

(a)

V_{IN} = 48 V DC with 12 V at 0 A

(b)

V_{IN} = 48 V DC with 12 V at 50 A

(c)

V_{IN} = 48 V DC with 12 V at 50 A

(d)

Figure 105 Output turn-on and turn-off with gate drive imbalance at V_{IN} = 48 V DC

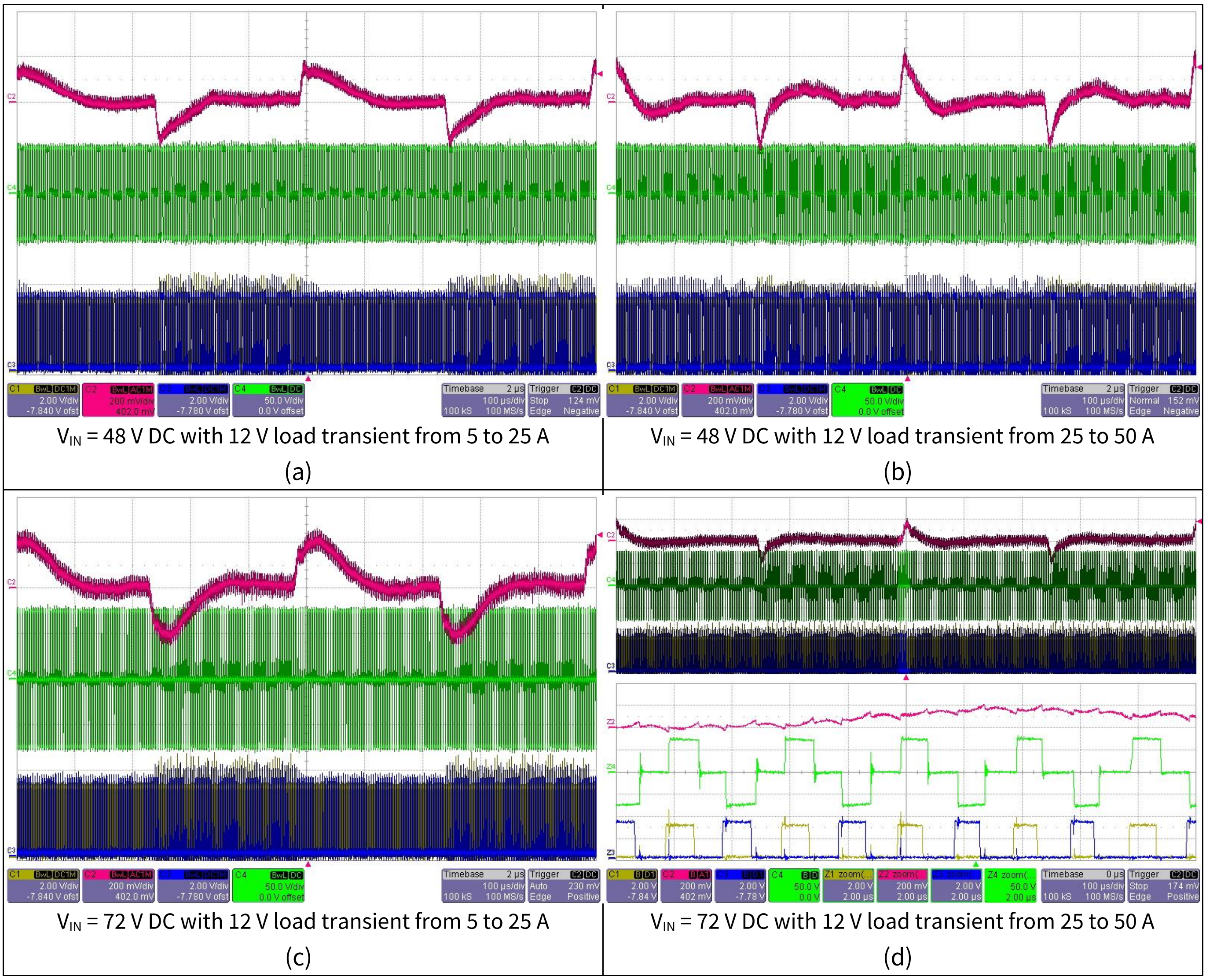
The start-up and shutdown waveforms are smooth and clean. No failure with the 30 ns mis-match is applied to the primary gate drive.

6.4.5 Load transient and flux balance

Output transient response is tested with the flux balance enabled. The primary-side gate drive has the 30 ns gate drive timing imbalance.

Ch1 = primary drive PWM1, Ch2 = V_{OUT}, Ch3 = primary drive PWM2, Ch4 = primary-side transformer winding

The e-load transition rate is set to 5 A/µs.



V_{IN} = 48 V DC with 12 V load transient from 5 to 25 A

(a)

V_{IN} = 48 V DC with 12 V load transient from 25 to 50 A

(b)

V_{IN} = 72 V DC with 12 V load transient from 5 to 25 A

(c)

V_{IN} = 72 V DC with 12 V load transient from 25 to 50 A

(d)

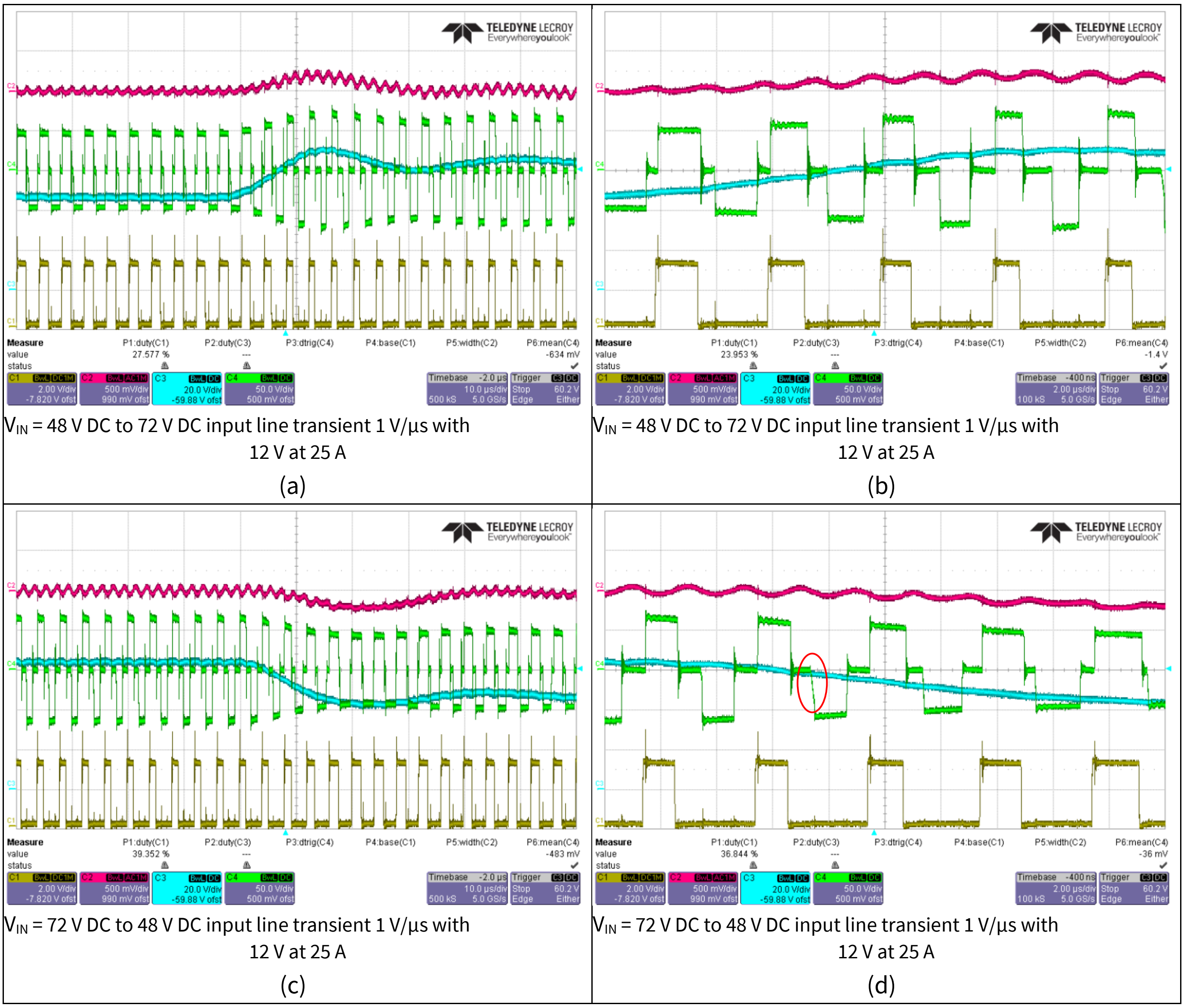
Figure 106 Output transient waveforms with flux balance enabled

It can be seen that during output transient with 30 ns primary gate drive timing imbalance, we do not see any imbalance in the voltage waveform taken across the primary transformer winding with the flux balance of the XDPP1100 enabled. In the detail waveforms during high-line full-load transient, there is no one-sided drop-out of the primary winding voltage.

6.4.6 V_{IN} transient and flux balance

Input line transient was applied to the converter and the switching waveforms were recorded.

Ch1 = primary drive PWM1, Ch2 = V_{OUT}, Ch3 = V_{IN}, Ch4 = primary-side transformer winding



V_{IN} = 48 V DC to 72 V DC input line transient 1 V/µs with 12 V at 25 A

(a)

V_{IN} = 48 V DC to 72 V DC input line transient 1 V/µs with 12 V at 25 A

(b)

V_{IN} = 72 V DC to 48 V DC input line transient 1 V/µs with 12 V at 25 A

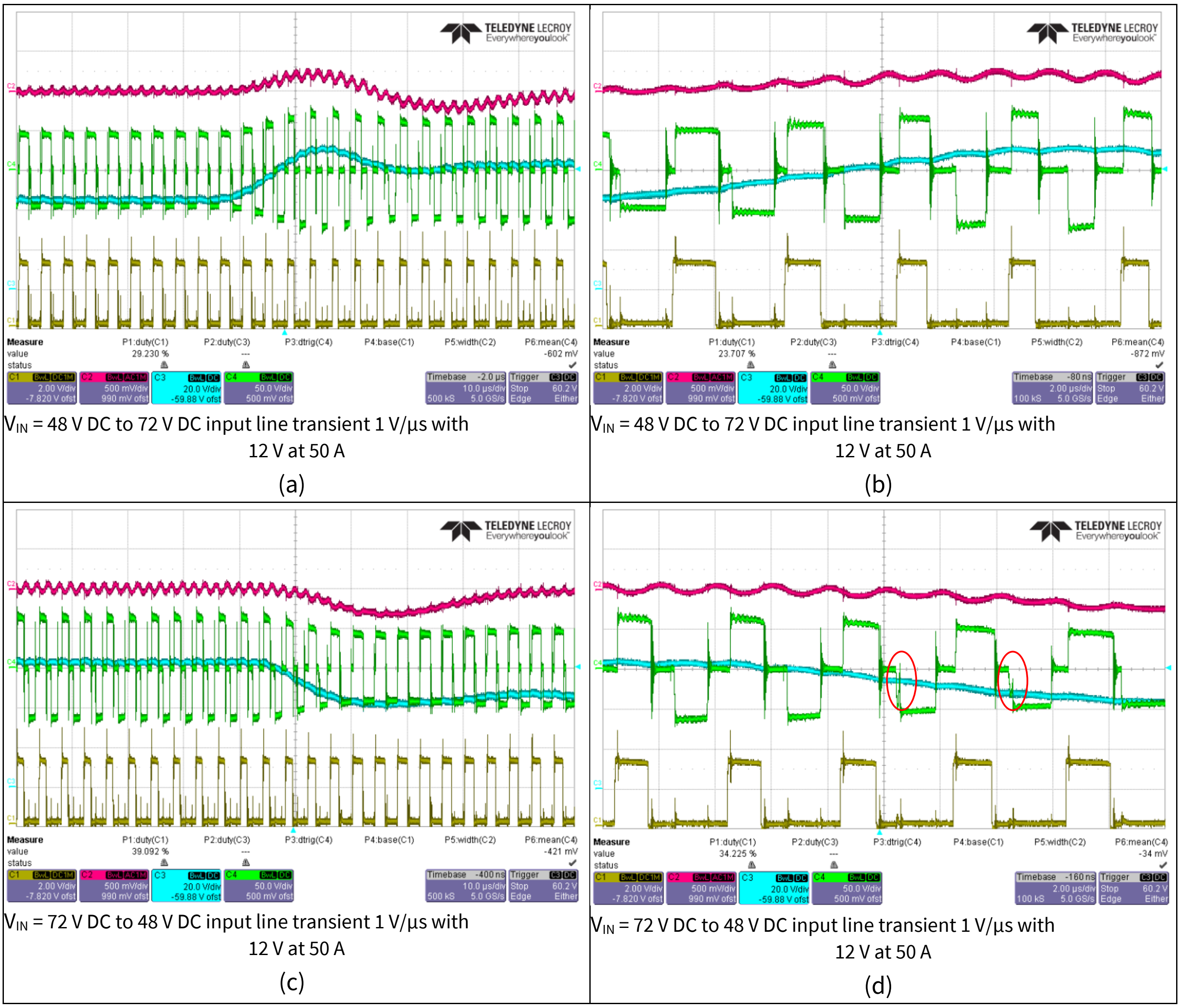
(c)

V_{IN} = 72 V DC to 48 V DC input line transient 1 V/µs with 12 V at 25 A

(d)

Figure 107 Flux balance during input line transient at half load

Ch1 = primary drive PWM1, Ch2 = V_{OUT}, Ch3 = V_{IN}, Ch4 = primary-side transformer winding



V_{IN} = 48 V DC to 72 V DC input line transient 1 V/µs with 12 V at 50 A

(a)

V_{IN} = 48 V DC to 72 V DC input line transient 1 V/µs with 12 V at 50 A

(b)

V_{IN} = 72 V DC to 48 V DC input line transient 1 V/µs with 12 V at 50 A

(c)

V_{IN} = 72 V DC to 48 V DC input line transient 1 V/µs with 12 V at 50 A

(d)

Figure 108 Flux balance during input line transient at full load

The figures show that during input line transient at the PWM off-time we see one- or two-cycle voltage ringing on one side of the transformer winding at half load and full load. The fact that this is only seen for one or two cycles shows the flux balance is working to correct the error during input line transient.

6.4.7 Flux balance fault protection

Table 50 is the configuration of the flux balance fault. The fault threshold should be set high enough to avoid tripping fault during line and load transient. Set **flux_bal_thresh** = 150 to enable proper operation.

Table 50 FB-FB VMC flux balancing fault configuration

Register name	Register value	Meaning
flux_bal_thresh	150	Flux balance fault threshold LSB = 2 volt-seconds (µs) Equation used: [(ve+vo)(te-to)+(te+to)(ve-vo)]/256
fbal_integ_thresh	0	Flux balance integrator error threshold defined with respect to maximum integrator range. 0 = disable
fbal_lpf_kpshift	2	Coefficient of LPF at output of fbal error
fbal_delta_abs_en	0	Determines whether absolute values to be applied to the error input before or after the LPF. 0 = convert to absolute value after LPF
fault_shut_mask_com	16_D	Fault interrupt shutdown converter at fbal_fault asserted

Figure 109 is the flux fault shutdown waveform. The **flux_bal_thresh** was triggered when load stepping from 25 A to 50 A. The primary gate drive has 30 ns timing mis-match. To reduce the sensitivity of the flux balance fault, increasing **fbal_lpf_kpshift** to 4 would not trigger fbal fault during transient.

Ch1 = primary drive PWM1, Ch2 = V_{OUT}, Ch3 = primary drive PWM2, Ch4 = transformer primary voltage.

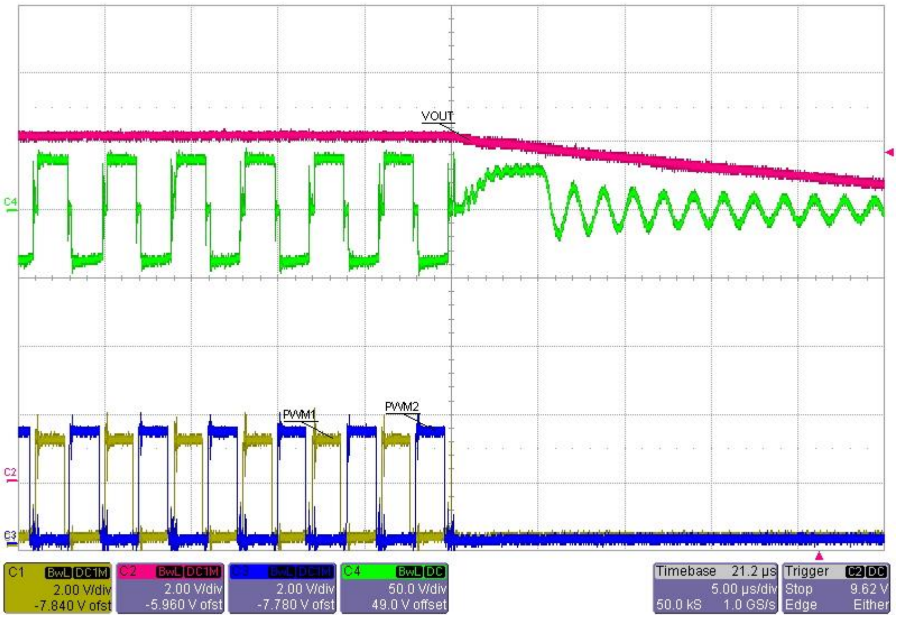


Figure 109 Converter shutdown on fbal fault (36 V, 10 V_{OUT}, 50 A load, flux_bal_thresh = 150)

7 Fast transient response

Load transient response is the capability of a power supply to respond to a sudden load change. Under a load transient condition, the output voltage will undershoot or overshoot before the controller takes action to bring the output back to the preset value. Output load transient is a critical performance measure of brick converters. Linear transient response per feedback loop is usually not fast enough. This chapter introduces the fast transient response (FTR) feature of the XDPP1100, demonstrates the performance compared to the linear PID response and discusses the benefits over the conventional approach.

7.1 FTR at positive load transient

The XDPP1100 implements FTR to HB and FB topologies in VMC. **Figure 110** is the schematic of a FB converter with full-wave synchronous rectification. The output voltage is sense by VSEN pin through a resistor divider connected to the output voltage. The resistor divider ratio determines the value of the PMBus command VOUT_SCALE_LOOP. The fast transient waveform is shown in **Figure 111**.

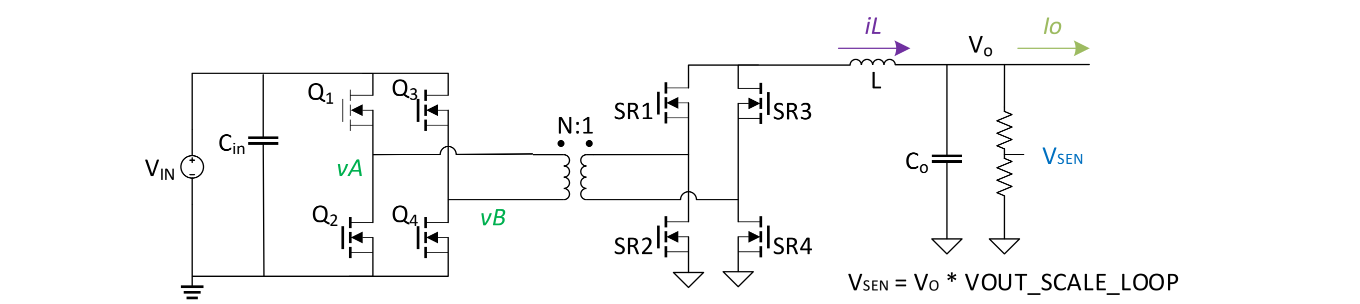


Figure 110 FB converter

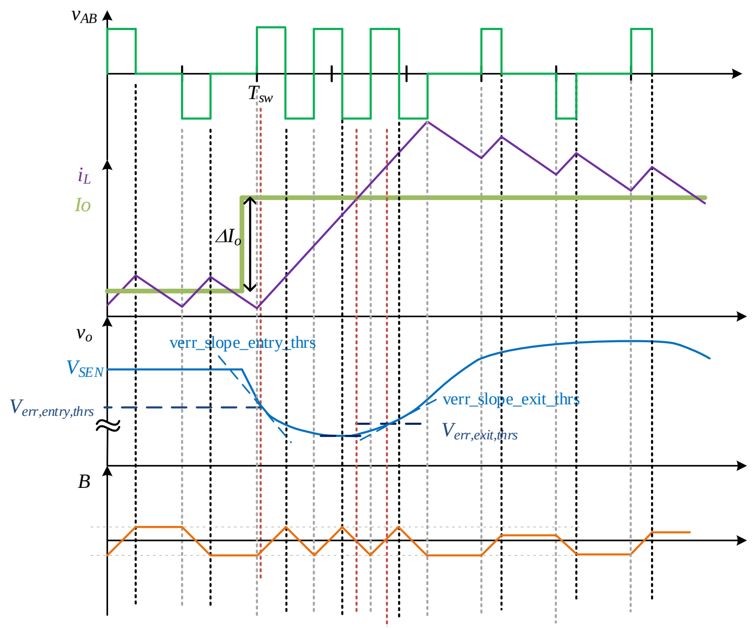


Figure 111 Waveforms of FTR with transformer protection

On detection of a positive load transient (I_o), the converter effectively maximizes the duty-cycle by changing the switching period from T_{sw} to $2T_{on}$ where T_{on} is the product of duty-cycle D and switching period T_{sw}. This makes the duty-cycle nearly 100 percent in FTR mode, with only dead-time applied to PWMs. The output inductor current (i_L) continuously rises and the converter maximizes the power delivered to the load. The voltage undershoot slope is reduced in FTR. Besides undershoot reduction, it can be seen that the transformer flux (B) is bounded. This is because the T_{on} time remains the same as in normal regulation. This provides the ability to optimize the transformer core without over-designing the transformer for load transient.

Figure 111 shows that output voltage starts to recover after a couple of FTR cycles, when the current in the output inductor is higher than the load current. To avoid overshoot, the converter should exit FTR mode before V_{OUT} returns to the set target voltage. The XDPP1100 exits FTR mode at a configurable error voltage threshold which is always below target V_{OUT}. The error voltage is defined as (target voltage - sense voltage). When the error voltage is smaller than the set threshold, the XDPP1100 exits FTR mode. At FTR exit, the XDPP1100 would complete the present FTR cycle then resume the switching period back to T_{SW}, and the linear loop then takes over for regulation.

In FTR mode, switching frequency is increased. Because converter duty-cycle D varies with input voltage, the operating frequency in FTR mode is also dependent on input voltage.

The fast transient entry and exit detections are achieved by threshold voltage detection at the VSEN pin, or by the derivative of the error voltage (voltage slope), or a combination of the two. Table 51 lists the threshold registers for positive load transient FTR. Design examples are provided in the table. For the dual-loop version of XDPP1100, independent FTR is provided: configure loop 0 FTR by pid0 registers and configure loop 1 FTR by pid1 registers.

Table 51 FTR registers for positive load transient (standard FTR)

Register name	Description
pid_verr_entry_thrs	FTR mode error voltage entry threshold. When (error voltage > pid_verr_entry_thrs) AND (error voltage slope > pid_verr_slope_entry_thrs) the control loop enters FTR mode. Notes: 1. This threshold is always positive, indicating that the controller enters FTR mode only when the sensed voltage is below the target. 2. Setting pid_verr_entry_thrs = 0 disables FTR mode. LSB = 1.25 mV, range = 0.0 to 158.75 mV at VSEN • Example: – pid_verr_entry_thrs = error voltage * VOUT_SCALE_LOOP/1.25 mV – V_{OUT} = 5 V, VOUT_SCALE_LOOP= 0.1683 – Enter FTR mode when V_{OUT} dips 45 mV below 5 V – pid_verr_entry_thrs = 45 mV * 0.1683/1.25 mV = 6
pid_verr_slope_entry_thrs	FTR mode error voltage slope entry threshold. When (error voltage > pid_verr_entry_thrs) AND (error voltage slope > pid_verr_slope_entry_thrs) the control loop enters FTR mode. Note: This threshold is always positive, indicating that the controller enters FTR mode as the sensed voltage is decreasing toward maximum undershoot. LSB = 1.25 mV/clk, range = 0.0 to 158.75 mV/clk at VSEN The clk is based on 50 MHz clock, equal to 2^7 x 20 ns = 2.56 µs. The slope is measured and calculated every three ADC code transitions (3 * 1.25 mV = 3.75 mV at VSEN). • Example: – To enter FTR at V_{OUT} slope higher than 20 mV/µs, with VOUT_SCALE_LOOP =

Register name	Description
	0.1683 – pid_verr_slope_entry_thrs = 20 mV/µs * VOUT_SCALE_LOOP/(1.25 mV/2.56 µs) = 6.9, set to 7
pid_verr_exit_thrs	FTR mode error voltage exit threshold. When (error voltage < pid_verr_exit_thrs) AND (error voltage slope < pid_verr_slope_exit_thrs) the control loop exits FTR mode. Note: This threshold is always positive, indicating that the controller exits FTR mode prior to the sensed voltage overshooting the target. LSB = 1.25 mV, range = 0.0 to 158.75 mV at VSEN • Example: – pid_verr_exit_thrs = error voltage * VOUT_SCALE_LOOP/1.25 mV – V_{OUT} = 5 V, VOUT_SCALE_LOOP = 0.1683 – Exit FTR mode when V_{OUT} is 25 mV below 5 V – pid_verr_exit_thrs = 25 mV * 0.1683/1.25 mV = 3
pid_verr_slope_exit_thrs	FTR mode error voltage slope exit threshold. When (error voltage < pid_verr_exit_thrs) AND (error voltage slope < pid_verr_slope_exit_thrs) the control loop exits FTR mode. Notes: 1. This threshold is always negative, indicating that the controller does not exit FTR mode until the sensed voltage has hit its maximum undershoot and is approaching the target voltage. 2. There is a -1.25 mV offset (i.e., code 0 = -1.25 mV, 1 = -2.5 mV, ...) LSB = -1.25 mV/clk, range = -1.25 to -160 mV/clk at VSEN The clk is based on 50 MHz clock, equal to 2^7 x 20 ns = 2.56 µs. The slope is measured and calculated every three ADC code transitions (3.75 mV at VSEN). • Example: – To exit FTR at V_{OUT} slope less than -10 mV/µs, with VOUT_SCALE_LOOP = 0.1683 – pid_verr_slope_exit_thrs = -10 mV/µs * VOUT_SCALE_LOOP/ (-1.25 mV/2.56 µs) - 1 = 2.4, set to 2
pid_ftr_lpf	FTR mode filter bandwidth on error voltage input: 0 = 1 MHz 1 = 2 MHz 2 = 4 MHz 3 = 8 MHz 4+ = filter bypassed

To avoid triggering FTR by switching noise on the VSEN pin, a LPF is added to the error sense circuit. The filter bandwidth can be selected from 1 MHz/2 MHz/4 MHz/8 MHz, or bypassed.

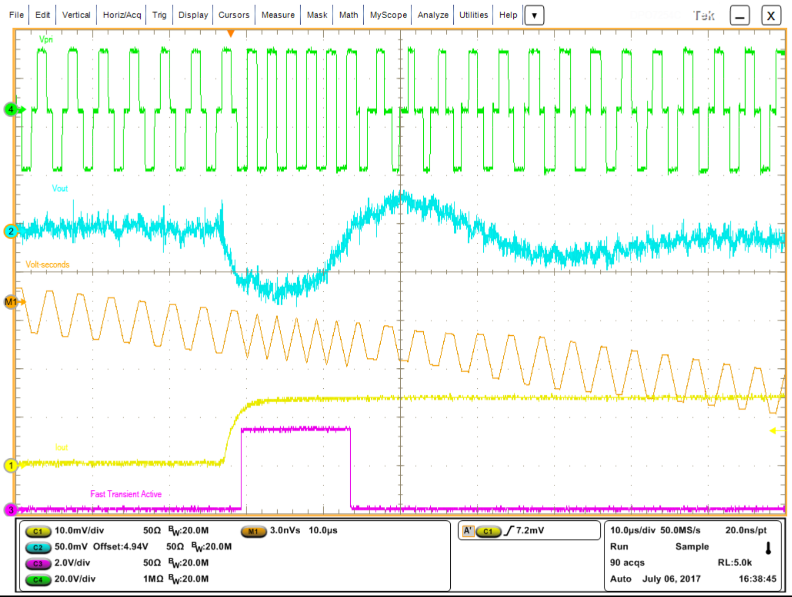
It is recommended to use the combination of voltage threshold and slope threshold for FTR entry and exit detection. With the help of slope detection, the exit voltage threshold could be set either lower or higher than the entry threshold, thus giving a wider adjustment range.

To enable the FTR feature, set register **pid_verr_entry_thrs** to a non-zero value and store the configuration into OTP. The FW will enable FTR when the converter is turned on. Please note that a read of the **pid_verr_entry_thrs** register will return 0 after the converter output is turned off. The value will be restored by FW during the next output enable. The **pid_verr_entry_thrs** could be adjusted during operation for the user to fine-tune the threshold. The value is not required to be stored into OTP to be effective, but a configuration that is not stored in OTP will be lost on resetting FW or power-cycling the V_{DD} of XDPP1100.

Figure 112 shows the experimental result of a HB converter with positive load transient from 0 A to 16.67 A. The output voltage is 5 V. VOUT_SCALE_LOOP = 0.1683. Transient loads were constructed by connecting two 5 W 0.15 Ω power resistors in series. The 0.3 Ω load represents a 16.67 A load at 5 V. The configurations of the FTR registers are listed in Table 52. The probe channels are defined in Table 53.

Table 52 Register settings for testing at 48 V

Register name	Register value (decimal)
pid0_verr_entry_thrs	6 (45 mV)
pid0_verr_slope_entry_thrs	8 (23.2 mV/µs)
pid0_verr_exit_thrs	3 (25 mV)
pid0_verr_slope_exit_thrs	3 (-11.6 mV/µs)



C1 10.0mV/div 50Ω BW:20.0M
C2 50.0mV Offset:4.94V 50Ω BW:20.0M
C3 2.0V/div 50Ω BW:20.0M
C4 20.0V/div 1MΩ BW:20.0M
M1 3.0nVs 10.0µs
A' C1 ∫ 7.2mV
10.0µs/div 50.0MS/s 20.0ns/pt
Run Sample
90 acqs RL:5.0k
Auto July 06, 2017 16:38:45

Figure 112 HB converter FTR waveform at 48 V, 0 A → 16.67 A

Table 53 Oscilloscope channel definition for 48 V measurements

Ch	Signal	Ch	Signal
1	Transient current [10 A/div]	4	Transformer primary voltage [20 V/div]
2	Output voltage [50 mV/div] (4.94 V offset)	M1	Transformer primary volt seconds [3 nVs/div]
3	Fast transient active signal [2 V/div]		

Note: The slope of M1 is an artifact of the scope sampling, and not a result of DC bias.

7.2 Transformer optimization with FTR

When designing a power supply transformer, selecting a magnetic core is always the first step. The core material is usually chosen by operating frequency and temperature range. Once the material is selected, the core size should be calculated based on the power rating, operating voltage and switching frequency. The choice of the core is usually based on the trade-off of cost, size and performance. By Faraday's law, the flux density ΔB is calculated:

$$\Delta B = \frac{\int V \cdot t}{N \cdot Ae} \tag{7.1}$$

Where $\int V \cdot t$ is the volt-second applied to the transformer primary winding, N is transformer primary turns and Ae is the core cross-section area.

Without FTR, the maximum PWM turn-on time during load transient could go up to the maximum duty-cycle limit. The flux density could double at high-line compare to the value at low-line, which introduces the risk of transformer saturation.

With FTR enabled, the volt-second at high-line is almost the same as at low-line. The transformer always operates under a controlled safety condition. In some cases, use of the next smaller core is possible for higher power density.

Figure 113 is the experimental result of a HB brick converter at positive load transient, from 5 A to 21.67 A at high-line 75 V input. The output voltage is 5 V, VOUT_SCALE_LOOP= 0.1683. Transient loads were constructed by connecting two 5 W 0.15 Ω power resistors in series. The 0.3 Ω load represents a 16.67 A load at 5 V. The configurations of the FTR registers are listed in Table 54. The probe channels are defined in Table 55.

Table 54 Register settings for testing at 75 V

Register name	Register value (decimal)
pid0_verr_entry_thrs	7 (52.5 mV)
pid0_verr_slope_entry_thrs	10 (29 mV/µs)
pid0_verr_exit_thrs	8 (60 mV)
pid0_verr_slope_exit_thrs	3 (-11.6 mV/µs)

Table 55 Oscilloscope channel definition

Channel	Signal
1	Transient current [10 A/div] Note: Zero is the static value 5 A.
2	Output voltage of FTR [50 mV/div] (5.0 V offset)
M1	Transformer flux density of FTR [75 mT/div]
4	Transformer primary voltage of FTR [40 V/div]
R2	Output voltage of linear PID [50 mV/div] (5.0 V offset)
R3	Transformer flux density of linear PID [75 mT/div]
R4	Transformer primary voltage of linear PID [40 V/div]

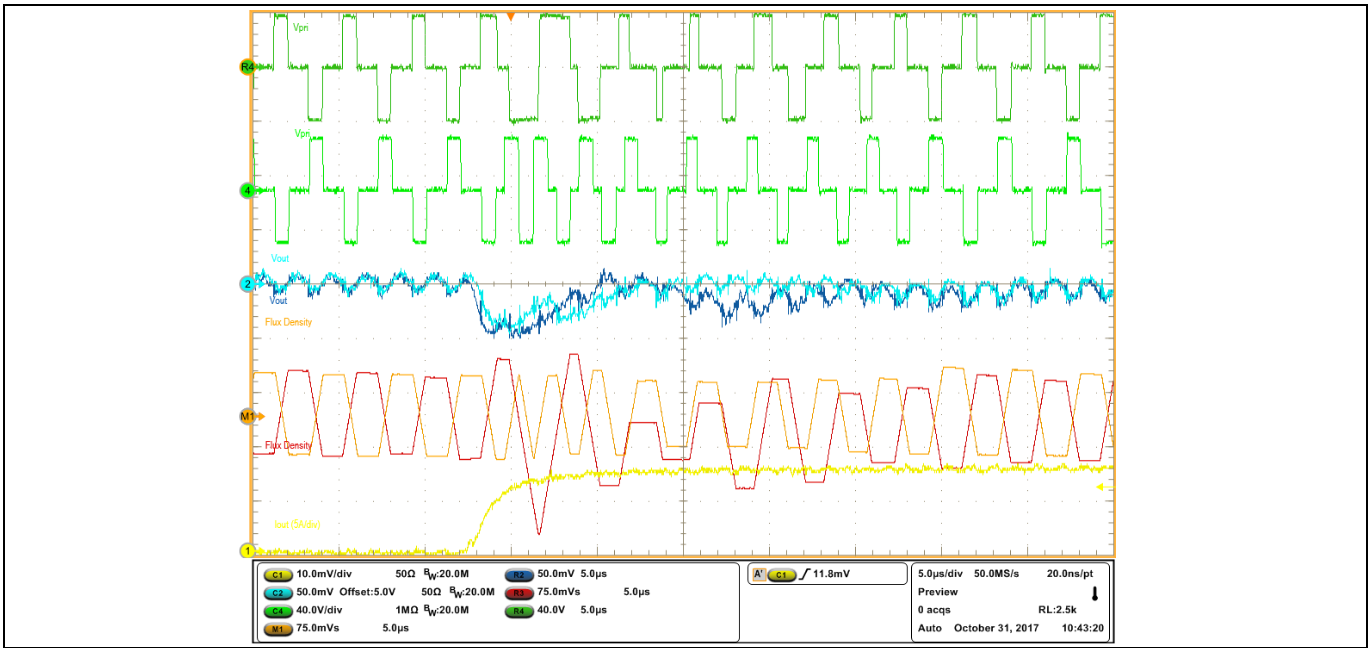


Figure 113 Fast transient vs. PID at 75 V (5 A → 21.7 A)

Figure 113 compares FTR and linear PID response. Notice that the linear PID saturates the duty-cycle (seen by the square wave of the transformer voltage, channel R4) during a load transient. The fast transient also effectively saturates the duty-cycle, but at higher frequency. The output voltage of both methods therefore experiences the same undershoot. The merit of fast transient is shown by the flux density (M1 vs. R3) where the peak-to-peak flux density is considerably greater when the linear PID is used. The peak-to-peak flux density is twice as large when the linear PID is used compared to fast transient.

With linear PID, the duty-cycle can be clamped during transients to avoid high flux density. The result is shown in **Figure 114**. While the flux is constrained, the output voltage undershoot is considerably larger than FTR.

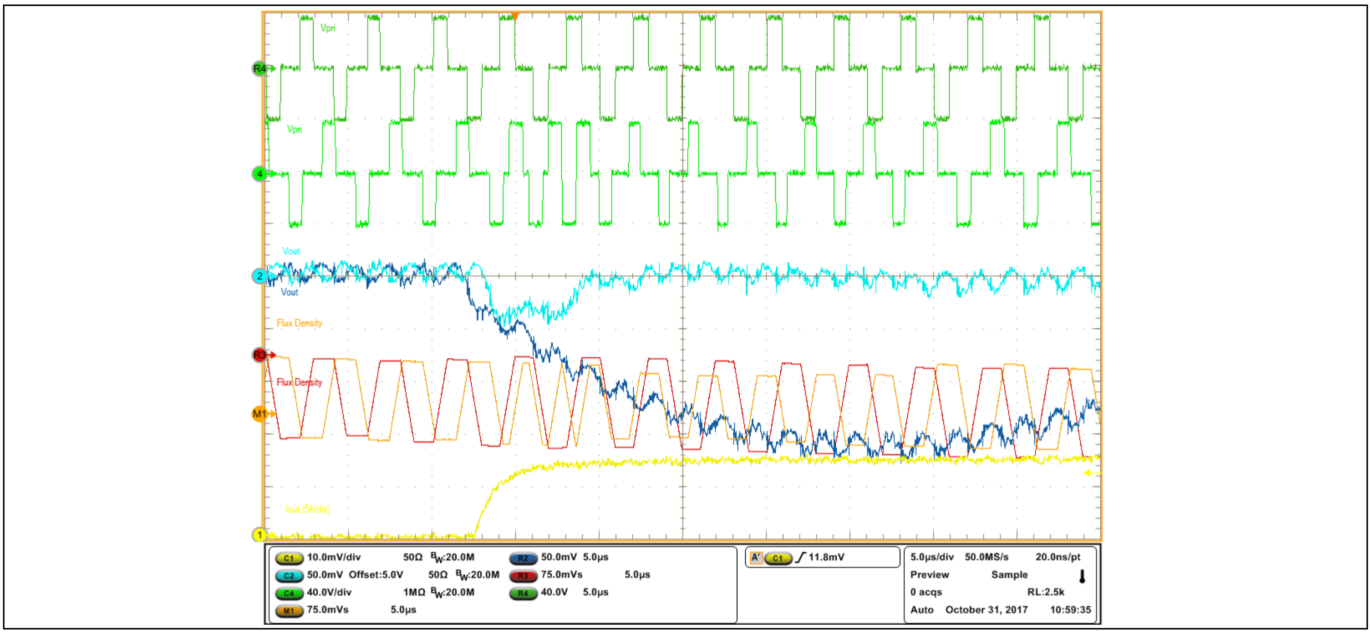


Figure 114 Fast transient vs. clamped duty-cycle PID at 75 V (5 A → 21.7 A)

7.3 Impact of output inductor and duty-cycle limit in FTR

High-current applications usually have a smaller output inductor. The current ripple in the output inductor is high, especially at the maximum input voltage. For example, a 48 V to 12 V/50 A FB converter has a 0.47 µH output inductor. At 48 V input, the inductor ripple current is 13 A; at 72 V input, the ripple current is 26 A. When the output inductor is very small, output voltage could overshoot during FTR mode. To optimize fast transient in such applications, a user-configurable maximum duty limit is added to the FTR.

The maximum duty-cycle limit is only activated when input voltage is higher than a set threshold (**lp_ftr_vin_thresh**). When the XDPP1100 enters FTR with input voltage higher than the threshold, the PWM pulse width is reduced per the **lp_ftr_vin_thresh** and input voltage ratio. The switching period is still $2T_{on}$ where T_{on} is the pulse width based on the feed-forward duty-cycle. This effectively limits the duty-cycle during FTR to lp_ftr_vin_thresh/V_{IN}, thus slowing down the increment of the inductor current (**Figure 116**).

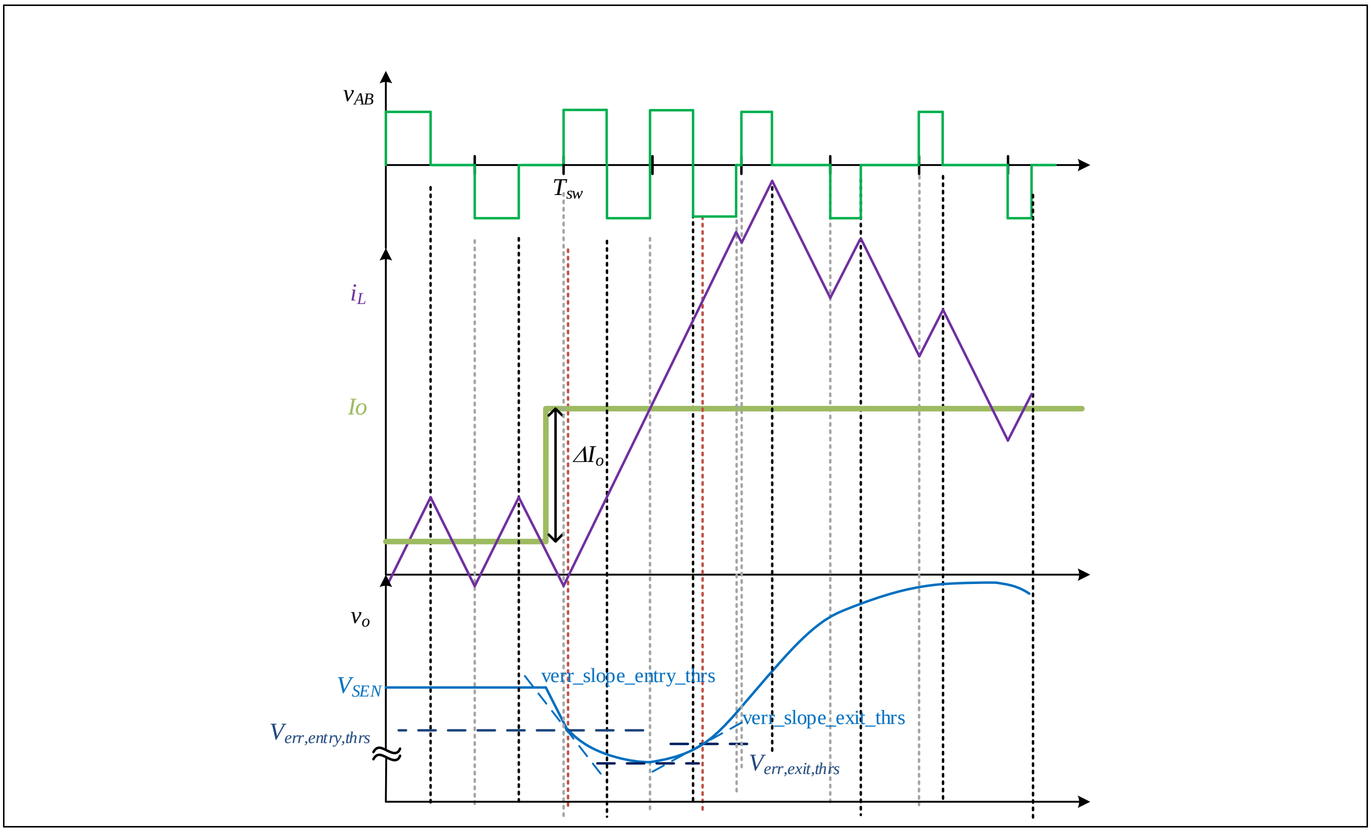


Figure 115 Voltage overshoot during FTR mode due to a small output inductor

Figure 115 shows the problem of overshoot after FTR during a positive load transient. **Figure 116** is the waveform of FTR with duty-cycle limit. With duty-cycle limit, voltage undershoot during load transient will be higher than without duty-cycle limit. But the overshoot could be greatly reduced, thus avoiding oscillation during fast transient.

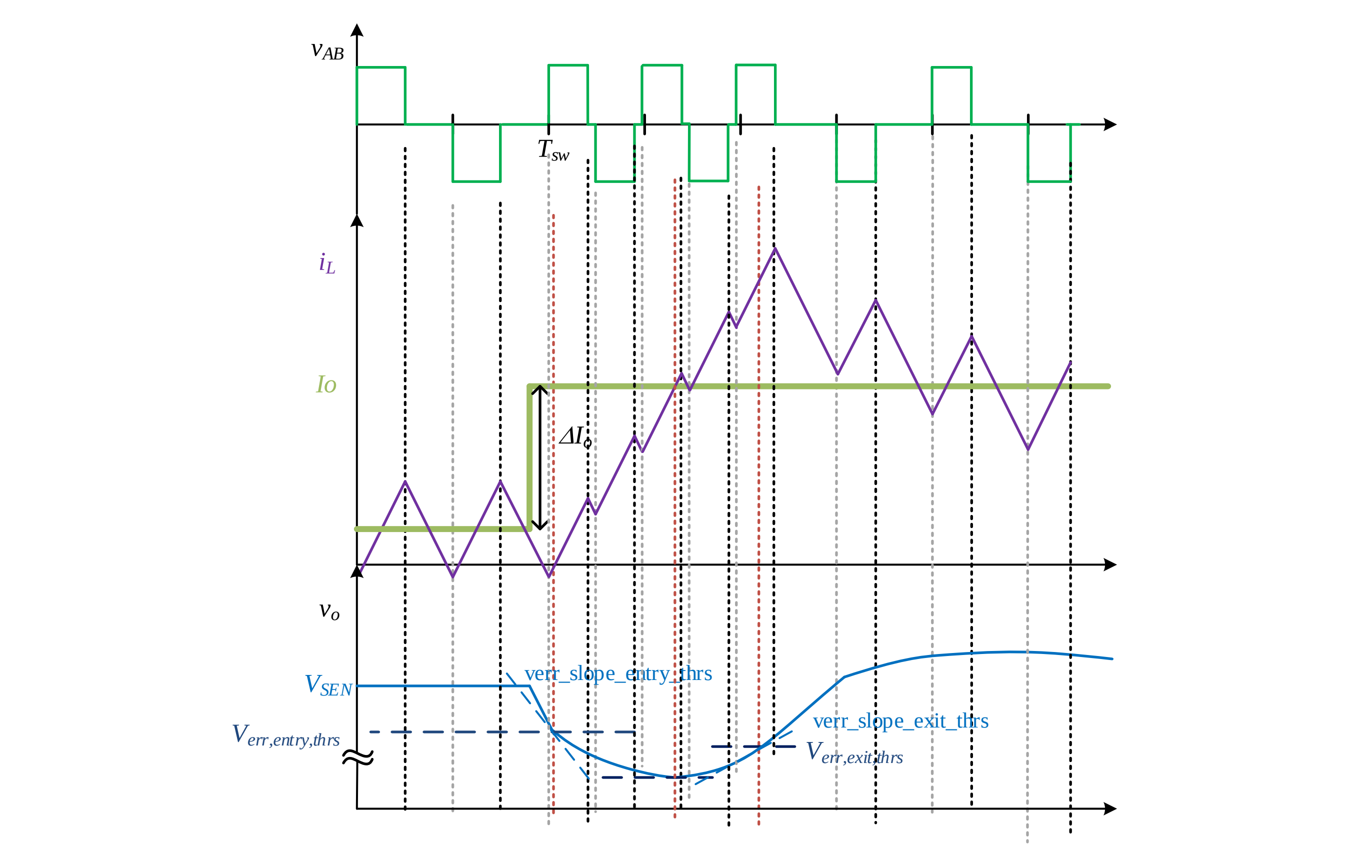


Figure 116 FTR with duty-cycle limit

Table 56 gives the register description and design example of **lp_ftr_vin_thresh**. V_{RECT} is the transformer secondary voltage after diode or SR MOSFET rectifiers (for more details of V_{RECT} voltage sensing, please refer to chapter 2.3). The amplitude of the V_{RECT} voltage is proportional to the input voltage V_{IN} and is scaled by the transformer turns ratio. In a bridge topology, V_{OUT}/V_{RECT} determines feed-forward duty-cycle.

Table 56 FTR input voltage threshold

Register name	Description
lp0_ftr_vin_thresh	Loop 0 FTR input voltage threshold. Below this threshold, the FTR PW is based on the feed forward duty-cycle from the PID. Above this threshold, the FTR PW is reduced in proportion to V_{IN}. If (V_{IN} < lp0_ftr_vin_thresh) FTR PW = (V_{OUT}/V_{RECT}) * $(T_{switch}/2)$ else FTR PW = (V_{OUT}/V_{RECT}) * $(T_{switch}/2)$ * (lp0_ftr_vin_thresh/V_{IN}) LSB 1 V, range 0 to 127 V • Example: – 48 V to 5 V HB power supply, transformer turns ratio Ns:Np = 1:3 – F_{sw} = 250 kHz, T_{switch} = 4 µs If set lp0_tr_vin_thresh = 80 (V) – At 58 V input voltage, FTR PW = 5 V/(58 V/2/3) * (4 µs/2) = 1.03 µs – Switching period in FTR = 2 * 1.03 µs = 2.06 µs

Register name	Description
	If set lp0_tr_vin_thresh = 48 (V) – At 58 V input, FTR PW = 5 V/(58 V/2/3) *(4 µs/2) * (48 V/58 V) = 0.856 µs – Switching period in FTR = 2 * 1.03 µs = 2.06 µs – The FTR duty-cycle = 2 * 0.856 µs/2.06 µs = 83 percent
lp1_ftr_vin_thresh	Loop 1 FTR input voltage threshold

Figure 117 shows a test waveform of a 5 V HBCT converter at load transient from 5 A to 15 A. Input voltage is 72 V. With **lp0_tr_vin_thresh** = 48 (V), the maximum duty-cycle during FTR is clamped to 67 percent by limiting PW to 0.56 µs. With **lp0_tr_vin_thresh** = 80 (V), the duty-cycle in FTR is 100 percent and PWM PW is 0.83 µs. The VOUT_SCALE_LOOP of this board is 0.23.

Table 57 Register settings for testing of FTR duty-cycle limit

Register name	Register value (decimal)
pid0_verr_entry_thrs	8 (43.5 mV)
pid0_verr_slope_entry_thrs	5
pid0_verr_exit_thrs	6 (33 mV)
pid0_verr_slope_exit_thrs	5
lp0_tr_vin_thresh	Test 1 = 48 V (enable duty-cycle limit in FTR) Test 2 = 80 V (disable duty-cycle limit in FTR)

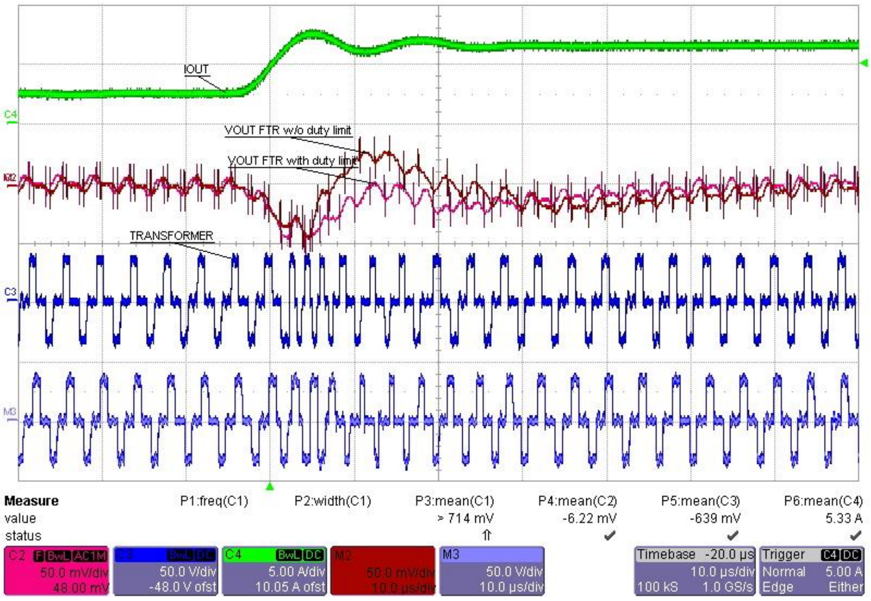
Ch4: transient load current [10 A/div]

Ch2: output voltage, FTR with duty-cycle limit [50 mV/div] (AC coupling)

M2: output voltage, FTR without duty-cycle limit [50 mV/div] (AC coupling)

Ch3: transformer primary voltage, FTR with duty-cycle limit [50 V/div]

M3: transformer primary voltage, FTR without duty-cycle limit [50 V/div]



Measure	P1:freq(C1)	P2:width(C1)	P3:mean(C1)	P4:mean(C2)	P5:mean(C3)	P6:mean(C4)
value			> 714 mV	-6.22 mV	-639 mV	5.33 A
status			⇑	✓	✓	✓

C2 F BwL AC1M 50.0 mV/div 48.00 mV
C3 BwL DC 50.0 V/div -48.0 V ofst
C4 BwL DC 5.00 A/div 10.05 A ofst
M2 50.0 mV/div 10.0 µs/div
M3 50.0 V/div 10.0 µs/div
Timebase -20.0 µs 10.0 µs/div 100 kS 1.0 GS/s
Trigger C4 DC Normal 5.00 A Edge Either

Figure 117 FTR waveform at 72 V input

7.4 FTR at load release

In the case of load release (negative load transient), on detecting the overshoot error threshold, the XDPP1100 enters overshoot FTR mode (OVS FTR) and terminates primary PWM immediately. Then it switches primary PWM at a minimum PW, which is configured by PMBus command MFR_MIN_PW. Output inductor current decays rapidly and the converter draws minimum power from the input. Output voltage overshoot is reduced. The XDPP1100 exits OVS FTR when the error voltage is higher than the exit threshold. The error voltage is defined as (target voltage - sense voltage), thus it is negative in the overshoot condition. After exiting FTR, the XDPP1100 resumes normal duty-cycle operation determined by the PID loop.

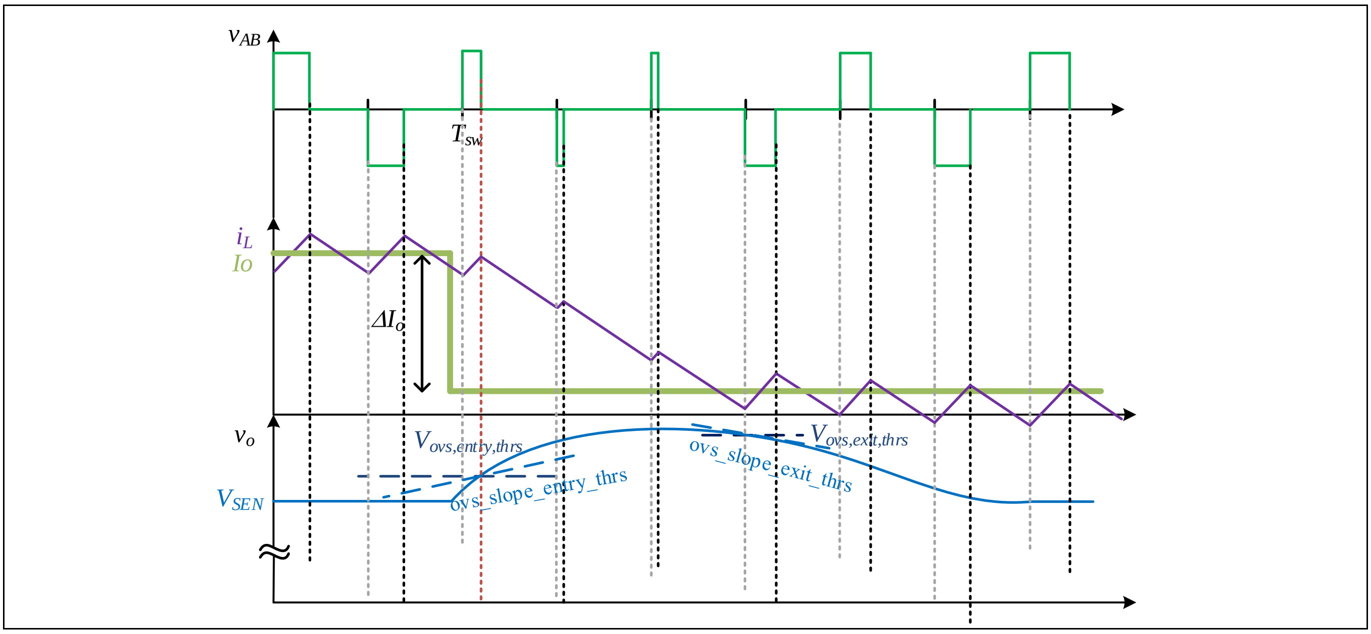


Figure 118 Fast transient waveform at load release

Table 58 lists the registers of overshoot FTR configuration. Similar to standard FTR, the overshoot FTR entry and exit detections are achieved by threshold voltage detection at the VSEN pin, or VSEN voltage slope detection, or a combination of the two. In the case of an exit threshold lower than the entry threshold (the absolute error voltage is bigger), it is suggested to use the combination of both error voltage threshold and slope threshold for exit detection, which could prevent the FTR exiting prematurely on the voltage rising slope.

FTR at load release uses the same LPF defined by **pid_ftr_lpf** (Table 51).

Table 58 FTR registers for load release (OVS FTR)

Register name	Description
pid_ovs_entry_thrs	Overshoot transient mode error voltage entry threshold. When (error voltage < pid_ovs_entry_thrs) AND (error voltage slope < pid_ovs_slope_entry_thrs) the control loop enters overshoot transient mode. Notes: 1. This threshold is always negative, indicating that the controller enters overshoot transient mode only when the sensed voltage is above the target. 2. Setting pid_ovs_entry_thrs = 0 disables overshoot transient mode. 3. There is a -1.25 mV offset (i.e., code 0 = -1.25 mV, 1 = -2.5 mV, ...) LSB = -1.25 mV, range = -1.25 to -160 mV at VSEN

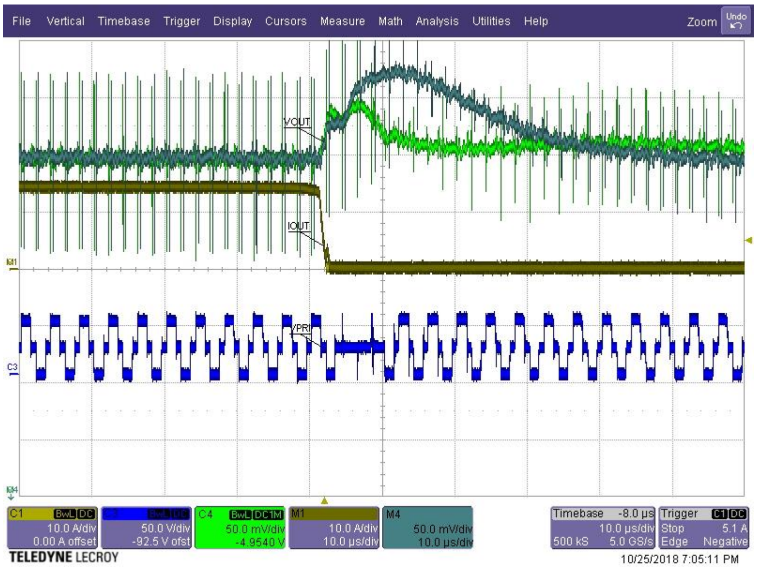
Register name	Description
	• Example: – pid_ovs_entry_thrs = error voltage * VOUT_SCALE_LOOP/(-1.25 mV) -1 – V_{OUT} = 5 V, VOUT_SCALE_LOOP = 0.1683 – Enter overshoot FTR mode when V_{OUT} overshoot is 30 mV above 5 V – pid_ovs_entry_thrs = (-30 mV) * 0.1683/(-1.25mV) - 1 = 3
pid_ovs_slope_entry_thrs	Overshoot transient mode error voltage slope entry threshold where the error voltage is defined as (target voltage - sense voltage). When (error voltage < pid_ovs_entry_thrs) AND (error voltage slope < pid_ovs_slope_entry_thrs) the control loop enters overshoot transient mode. Notes: 1. This threshold is always negative, indicating that the controller enters overshoot transient mode as the sensed voltage is increasing toward maximum overshoot. 2. There is a -1.25 mV offset (i.e., code 0 = -1.25 mV, 1 = -2.5 mV, ...) LSB = -1.25 mV/clk, range = -1.25 mV/clk to -160 mV/clk at VSEN The clk is based on 50 MHz clock, equal to 2^7 x 20 ns = 2.56 µs. The slope is measured and calculated every three ADC code transitions (3.75 mV at VSEN). • Example: – To enter overshoot FTR at overshoot slope lower (faster) than -6 mV/µs, with VOUT_SCALE_LOOP = 0.1683 – pid_ovs_slope_entry_thrs = (-6 mV/µs) * VOUT_SCALE_LOOP/ (-1.25 mV/2.56 µs) -1 = 1.07, set to 1
pid_ovs_exit_thrs	Overshoot transient mode error voltage exit threshold where the error voltage is defined as (target voltage - sense voltage). When (error voltage > pid_ovs_exit_thrs) AND (error voltage slope > pid_ovs_slope_exit_thrs) the control loop exits overshoot transient mode. Notes: 1. This threshold is always negative, indicating that the controller exits overshoot transient mode prior to the sensed voltage undershooting the target. 2. There is a -1.25 mV offset (i.e., code 0 = -1.25 mV, 1 = -2.5 mV, ...) LSB = -1.25 mV, range = -1.25 to -160 mV at VSEN • Example: – pid_ovs_exit_thrs = error voltage * VOUT_SCALE_LOOP/(-1.25 mV) -1 – V_{OUT} = 5 V, VOUT_SCALE_LOOP = 0.1683 – Exit overshoot FTR mode when V_{OUT} overshoot is less than -60 mV – pid_ovs_exit_thrs = (-60 mV) * 0.1683/(-1.25 mV) -1 = 7
pid_ovs_slope_exit_thrs	Overshoot transient mode error voltage slope exit threshold where the error voltage is defined as (target voltage - sense voltage). When (error voltage > pid_ovs_exit_thrs) AND (error voltage slope > pid_ovs_slope_exit_thrs) the control loop exits overshoot transient mode. Note: This threshold is always positive, indicating that the controller does not exit

Register name	Description
	overshoot transient mode until the sensed voltage has hit its maximum overshoot and is approaching the target voltage. LSB = 1.25 mV/clk, range = 0.0 to 158.75 mV/clk at VSEN The clk is based on 50 MHz clock, equal to 2^7 x 20 ns = 2.56 µs. The slope is measured and calculated every three ADC code transitions (3.75 mV at VSEN). • Example: – To exit overshoot FTR at overshoot slope higher than 6 mV/µs, with VOUT_SCALE_LOOP = 0.1683 – pid_ovs_slope_exit_thrs = 6 mV/µs * VOUT_SCALE_LOOP/(1.25 mV/2.56 µs) = 2.07, set to 2

Figure 119 is an example waveform of overshoot FTR. Tested with HBCT converter, 48 V input, 5 V output, load stepping from 16.67 A to 0 A. The VOUT_SCALE_LOOP of this board is 0.23. The register configuration is shown in Table 59. Channel 4 is the output voltage waveform with FTR. M4 is the output voltage waveform without FTR. The overshoot at load release was cut from 80 mV to 50 mV with fast transient.

Table 59 Register settings for testing of load release

Register name	Register value (decimal)
pid0_ovs_entry_thrs	9 (-54 mV)
pid0_ovs_slope_entry_thrs	2 (-6.4 mV/µs)
pid0_ovs_exit_thrs	8 (-49 mV)
pid0_ovs_slope_exit_thrs	2 (4.2 mV/µs)



M4: V_{OUT} without FTR (50 mV/div)

Ch4: V_{OUT} with FTR enabled (50 mV/div)

Ch1: output transient current (10 A/div)

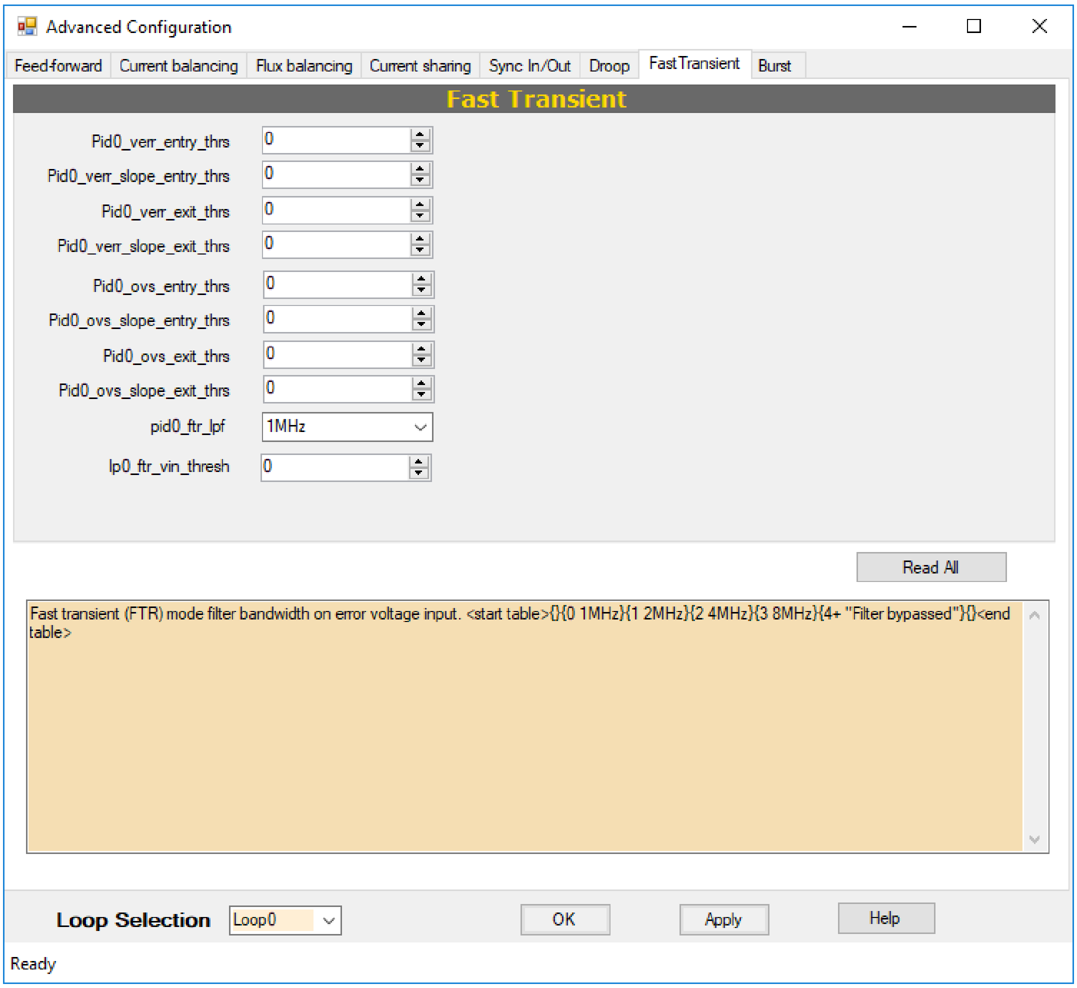
Ch3: primary switching node voltage (50 V/div)

Figure 119 Waveforms of FTR when load releases

Note: Both overshoot and undershoot fast transient are disabled during start-up and only activated when the output voltage reaches the regulation target voltage.

7.5 XDPP1100 GUI design tool for FTR

The XDPP1100 GUI design tool for fast transient configuration is shown in **Figure 120**.



Pid0_verr_entry_thrs 0
Pid0_verr_slope_entry_thrs 0
Pid0_verr_exit_thrs 0
Pid0_verr_slope_exit_thrs 0
Pid0_ovs_entry_thrs 0
Pid0_ovs_slope_entry_thrs 0
Pid0_ovs_exit_thrs 0
Pid0_ovs_slope_exit_thrs 0
pid0_ftr_lpf 1MHz
lp0_ftr_vin_thresh 0

Fast transient (FTR) mode filter bandwidth on error voltage input. <start table>{}{0 1MHz}{1 2MHz}{2 4MHz}{3 8MHz}{4+ "Filter bypassed"}{}<end table>

Loop Selection Loop0

Figure 120 GUI design tool – fast transient

7.6 Summary of FTR

The XDPP1100 fast transient offers many advantages to the user:

- It allows the user to optimize the transformer core as it maintains the same volt-seconds as in the steady-state even under the dynamic load conditions.
- Smaller undershoot during positive load transient without increasing flux density.
- Smaller overshoot during negative load transient.
- Ability to adjust FTR for different output chokes.
- Most importantly, it saves the user effort when optimizing the control loop for dynamic load conditions.

8 Current balancing

The current balancing of the XDPP1100 refers to balancing the phase current of an interleaved converter. Figure 121 is the schematic of interleaved HBCT topology. Phase 1 current is sensed by the ISEN/IREF pair and phase 2 current is sensed by the BISEN/BIREF pair.

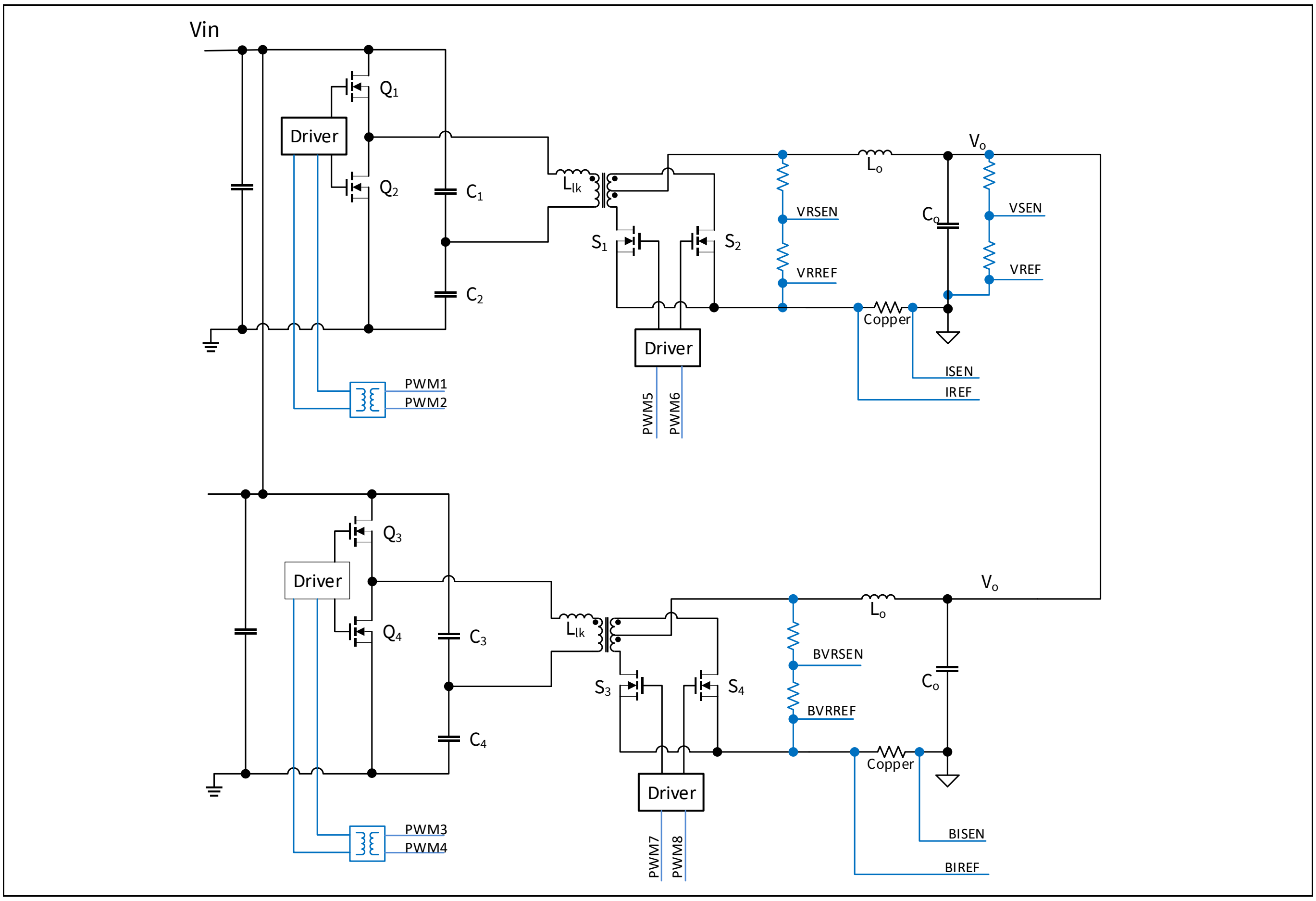


Figure 121 Interleaved HBCT converter

Figure 122 is the block diagram of the current balancing circuit. It subtracts phase 1 current from phase 2 to obtain the current difference between phase 1 and phase 2. The result is filtered and clamped then sent to phase1 to adjust duty-cycle in every switching frequency cycle. If phase 1 current is lower than phase 2 current, the ibal_duty_adj_o is positive and it increases duty-cycle of phase 1 until phase 1 current is equal to phase 2. Vice-versa, if phase 1 current is higher than phase 2, phase 1 duty-cycle is reduced to reduce the phase current until it equals phase 2 current.

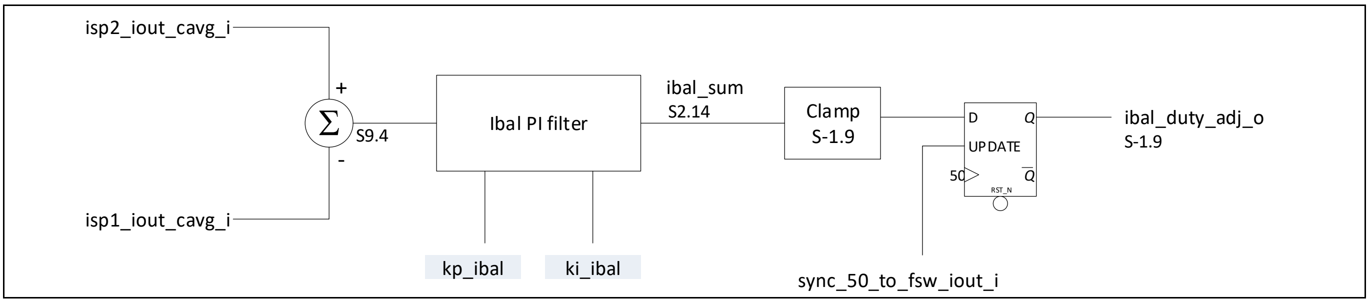


Figure 122 Current balancing block diagram

The PI filter is configured by register **kp_ibal** and **ki_ibal**.

Table 60 Current balancing PI filter register

Register	Description	Equation

kp_ibal	Current balancing PI filter proportional coefficient index	kp_exp = kp_ibal [5:3]
		kp_man = 8 + kp_ibal [2:0]
		kp = kp_man * 2^(kp_exp – 14)

ki_ibal	Current balancing PI filter integral coefficient index	ki_exp = ki_ibal [5:3]
		ki_man = 8 + ki_ibal [2:0]
		ki = ki_man * 2^(ki_exp – 20)

The magnitude response of the PI filter is defined by:

$$\sqrt{kp^2 + (\frac{ki}{2\pi \cdot Tsw \cdot f})^2} \qquad (8.1)$$

Here, T_{SW} is the converter switching period.

The current balancing circuit is enabled by FW when converter topology is interleaved topology and the bit [8] of PMBus command FW_CONFIG_REGULATION, INTERLEAVE_ENABLE is set to 1.

A dead-band can be set to disable current balancing when the total output current of the two phases is less than the threshold **ibal_en_thresh**. This threshold helps to avoid bad balancing when the current sense accuracy is low at light load, which is usually caused by offset error or due to the board's lack of capability to sense negative current.

Table 61 Ibal_en_thresh configuration table

Value	Current balancing enable threshold
0, 1	Always enabled for interleaved topology
2	3 A
3	5 A

8.1 Current balance registers description

Table 62 Current balance registers

Name	Address	Bits	Description
Common peripheral			
kp_ibal	7000_3000_H	[5:0]	Current balance PI filter proportional coefficient index. Set to 0 to disable the proportional component of the filter. Note that index settings greater than 55 are clamped to 55.
ki_fbal	7000_3000_H	[11:6]	Current balance PI filter integral coefficient index. Set to 0 to disable the integral component of the filter. Note that index settings greater than 55 are clamped to 55.
ibal_en_thresh	7000_3000_H	[27:26]	Total current level above which current balancing is enabled in interleaved (multiphase) topologies. Set to higher value if concerned about low current accuracy.
en_ibal	7000_3054_H	[2]	FW-driven HW block enabled for current balance function.

Name	Address	Bits	Description
ibal_fw_adj	7000_3078$_H$	[7:0]	When ibal_fw_en is high, this register overrides the HW current balance duty-cycle adjust output with a FW controlled setting. LSB = 0.195 percent, range = 0.0 to 24.805 percent
ibal_fw_en	7000_3078$_H$	[8]	Enables FW controlled current balance loop via ibal_fw_adj.

8.2 Current balance PMBus command

The current balance is enabled when the FW_CONFIG_REGULATION bit [8], INTERLEAVE_ENABLE is set to 1.

Please note, the interleaved topology is a single loop topology. The parameters should be configured in loop 0. On the other hand, a few PMBus commands in loop 1 should also be defined for proper interleaved operation, these commands are listed in Table 63.

Table 63 PMBus commands of Loop 1 for interleaved topology

Command name	Command code	Format	Description
MAX_DUTY	32$_H$	LINEAR11	Set the maximum duty-cycle in percentage. In interleaved topology, this parameter defines the maximum duty-cycle of ramp 1. It should be set to the same value as the MAX_DUTY of loop 0.
FREQUENCY_SWITCH	33$_H$	LINEAR11	Set the switching frequency in kHz. In interleaved topology, this parameter defines the frequency of ramp 1. It should be set to the same value as the FREQUENCY_SWITCH of loop 0.
INTERLEAVE	37$_H$		Set the phase shift of ramp 1. The recommended configuration for Loop 1: Group identification number = 1, number in group = 2, interleave order = 1. Loop 0 should configured the INTERLEAVE command as: Group identification number = 1, number in group = 2, interleave order = 0.
MFR_IOUT_APC	EA$_H$	LINEAR11	Set the IOUT APC gain for phase 2. Typically the value is the same as loop 0, but might need fine adjustment to trim the accuracy especially when using PCB copper as current shunt.

9 Burst operation

Burst operation is defined as a light load operating state during which the controller provides only a sequence of PWM bursts to reduce the switching losses of the converter and achieve low standby power. The XDPP1100 allows the user to configure the load current threshold below which it enters the burst mode, and to configure the desired burst mode voltage error threshold and the number of bursts. The frequency of the PWM in burst mode will be same as the switching frequency. The SR MOSFETs are turned off during burst mode to improve light load efficiency. The burst mode only applies to bridge topologies.

Figure 123 demonstrates the XDPP1100 burst mode operation.

- Burst mode operation is enabled by setting PMBus command POWER_MODE = 0.
- The XDPP1100 enters burst mode when the output current is lower than the burst entry threshold **pid_burst_mode_ith**.
- In burst mode, the SR MOSFETs are turned off.
- The primary PWM turns on when the output error voltage is higher than the **pid_burst_mode_err_thr** threshold. The error voltage is defined as (target_voltage – sensed_voltage). This mode is called burst-on mode. The SR gate drive stays off in the burst-on phase. The i-term of the PID is frozen while in burst mode, and the primary PWM duty-cycle is determined by the feed-forward duty-cycle.
- During burst, the primary PWM turns off after switching a pre-defined number of cycles, which is configured by **pid_burst_reps**. This mode is called burst-off mode. Both the primary PWM and SR are turned off in the burst-off phase.
- The XDPP1100 exits burst mode when the burst-off time is shorter than half of the switching period ($T_{sw}/2$). Primary PWM and SR gate drive resume normal operation with the duty-cycle that is determined by the PID loop.
- After the XDPP1100 exits burst mode, it will only re-enter after (2^15 x 1/f_{sw}) seconds.
- The XDPP1100 burst mode is only supported in bridge topologies.

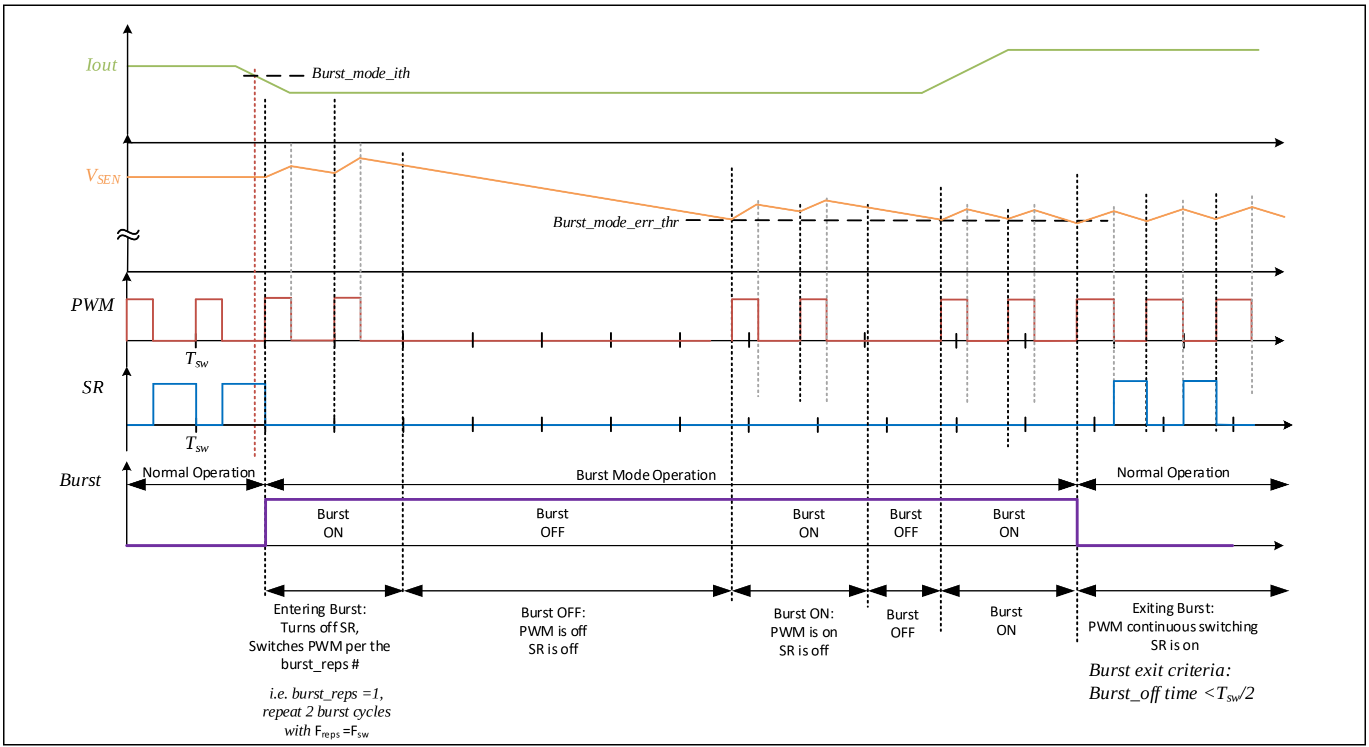

Iout
Burst_mode_ith
V_SEN
Burst_mode_err_thr
PWM
T_sw
SR
T_sw
Burst
Normal Operation
Burst Mode Operation
Normal Operation
Burst ON
Burst OFF
Burst ON
Burst OFF
Burst ON
Entering Burst: Turns off SR, Switches PWM per the burst_reps #
i.e. burst_reps =1, repeat 2 burst cycles with F_reps =F_sw
Burst OFF: PWM is off SR is off
Burst ON: PWM is on SR is off
Burst OFF
Burst ON
Exiting Burst: PWM continuous switching SR is on
Burst exit criteria: Burst_off time <T_sw/2

Figure 123 XDPP1100 burst mode operation

9.1 Configuring burst mode operation

This section describes the procedure to configure burst mode using the XDPP1100 controller.

- Determine the allowable burst mode output voltage ripple and load current threshold to enter burst mode.
- Configure the burst mode output voltage ripple threshold using register **pidX_burst_mode_err_thr**. The X in the register name denotes 0 for for loop 0 and 1 for loop 1.

$$Vout\_burst\_error = \frac{\text{pid\_burst\_mode\_err\_thr x 1.25mV}}{VOUT\_SCALE\_LOOP} \tag{9.1}$$

- Configure the load current threshold for burst mode using register **pidX_burst_mode_ith**.

$$Iout\_burst\_entry\_threshold = \frac{\text{pid\_burst\_mode\_ith x MFR\_IOUT\_APC}}{2} \tag{9.2}$$

- The number of PWM switching cycles in the burst-on phase can be configured using register **pidX_burst_reps**. The maximum number of bursts allowed is eight for a value of 3 and can only be configured in increments to the power of two, i.e. number of bursts = 2^(value in pid_burst_reps).
- Configure PMBus command 0x34, i.e. power mode as 0 (burst mode enabled) from 3 (normal mode).

9.1.1 Burst mode registers

Table 64 Burst mode registers

Name	Address (loop 0/1)	Bits	Description
PID 0/1 peripheral			
pid_burst_mode_err_thr	7000_1C14$_H$ 7000_2014$_H$	[3:0] [3:0]	Burst mode error voltage threshold where the error voltage is defined as (target voltage - sense voltage). When the controller is in burst mode, (error voltage > pid_burst_mode_err_thr) will trigger the start of a new burst sequence. This threshold is always positive, indicating that the controller triggers the start of a new burst sequence at or below the target voltage. LSB = 1.25 mV, range = 0 to 18.75 mV
pid_burst_mode_ith	7000_1C14$_H$ 7000_2014$_H$	[9:4] [9:4]	Burst mode entry current threshold. When burst mode is enabled (POWER_MODE [7:0] = 0x00), the controller will enter burst mode on the sensed current dropping below pid_burst_mode_ith. LSB = (QADC/2) where QADC is the value of MFR_IOUT_APC in amps Range = 0 to 31.5 QADC
pid_burst_reps	7000_1C14$_H$ 7000_2014$_H$	[11:10] [11:10]	Burst mode cycle count. In burst mode, one cycle corresponds to one even half-cycle pulse followed by one odd half-cycle pulse. This register defines the number of burst cycles in each burst event. A higher cycle count can be used to increase the inductor peak current in a burst event, which will increase the time between burst events at a given load current. 0 = 1 cycle

Name	Address (loop 0/1)	Bits	Description
			1 = 2 cycles 2 = 4 cycles 3 = 8 cycles

9.1.2 Burst mode PMBus commands

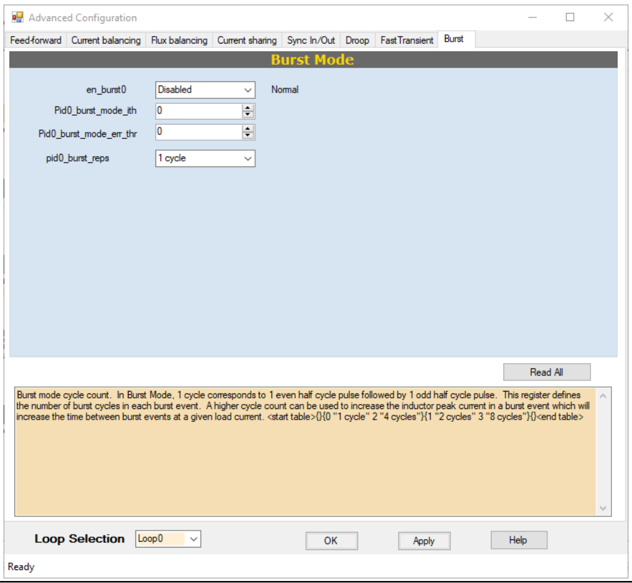
Table 65 shows the PMBus commands relevant to burst mode configuration.

Table 65 Burst mode – relevant PMBus commands

Command name	Command code	Format	Description
VOUT_SCALE_LOOP	29_H	DIRECT, U0.16	The VOUT_SCALE_LOOP command scales VOUT_COMMAND, etc. for the external resistor divider on the voltage sense input to the device.
POWER_MODE	34_H		Set the power conversion mode of operation: 0 = max. efficiency (enable burst mode) 3 = max. power
MFR_IOUT_APC	EA_H	LINEAR11, U1.9/U2.8	Current sense amps per code. Linear11 format amps/code unit.

9.1.3 GUI design tool for burst mode configuration

The XDPP1100 GUI provides a design tool to configure burst mode. The tool can be found in the "Advanced Features". The **en_burst** configuration has a drop-down list. By selecting "Enabled" burst mode, the correspond loop writes PMBus POWER_MODE = 0.



en_burst0: Disabled — Normal
Pid0_burst_mode_ith: 0
Pid0_burst_mode_err_thr: 0
pid0_burst_reps: 1 cycle

Burst mode cycle count. In Burst Mode, 1 cycle corresponds to 1 even half cycle pulse followed by 1 odd half cycle pulse. This register defines the number of burst cycles in each burst event. A higher cycle count can be used to increase the inductor peak current in a burst event which will increase the time between burst events at a given load current. <start table>{}{0 "1 cycle" 2 "4 cycles"}{1 "2 cycles" 3 "8 cycles"}{}<end table>

Loop Selection: Loop0

Ready

Figure 124 GUI design tool for burst mode

9.2 Entry and exit criteria of burst mode

This section explains the entry and exit criteria of burst mode operation with the help of some practical waveforms. A test was performed using a single loop (loop 0) HBCT converter (**Figure 125**). The threshold values used are as shown below.

Table 66 Burst control register values

Register name	Register value	Description
pid0_burst_mode_err_thr	10	Set burst error (ripple) voltage to 54.49 mV
pid0_burst_mode_ith	45	Set burst entry current threshold to 5.097 A
pid0_burst_reps	3	Every burst-on phase switches 8 cycles
VOUT_SCALE_LOOP	0.2294159	
MFR_IOUT_APC	0.224609	

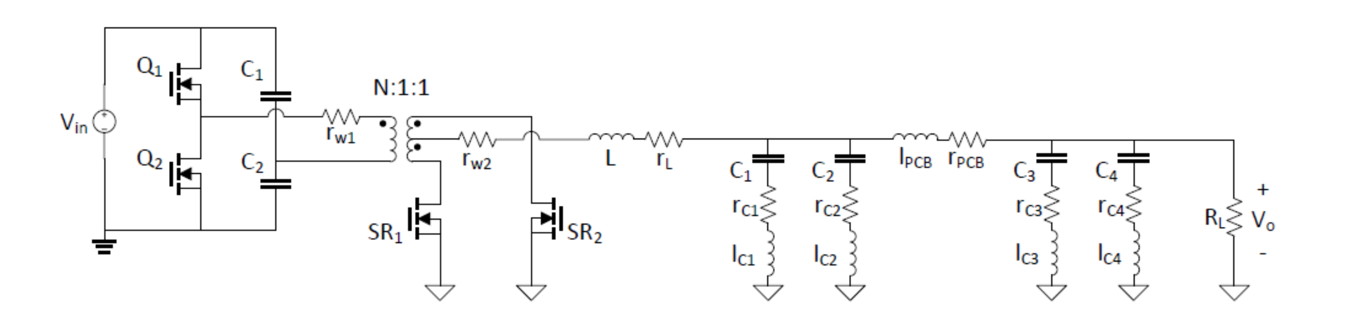
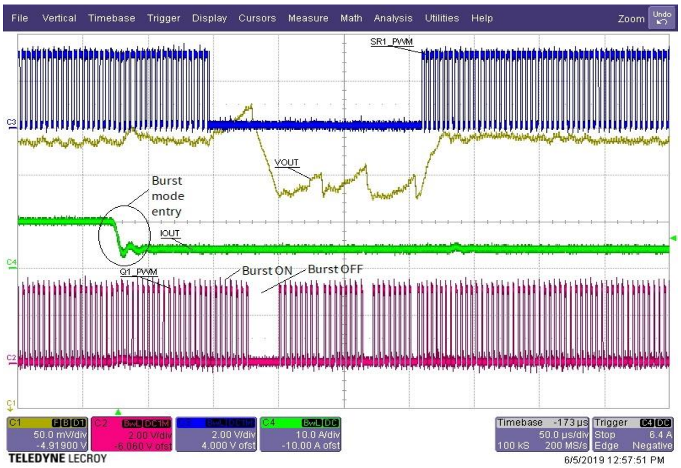


Figure 125 HBCT schematic

9.2.1 Burst mode entry criteria

As shown in **Figure 126**, the converter enters burst mode as the load current falls from 10 A to 4 A, which is less than the threshold 5.097 A, and burst comes up based on the voltage error limit at 63.5 mV, which is close to the error threshold of 54.49 mV. The number of bursts was set to 8, i.e. the value of burst_reps is 3. The accuracy of burst mode entry depends on the accuracy of the I_{OUT} telemetry of the system. The channels used in the waveform are described below.



Ch3: SR1 PWM, 2 V/div, PWM control signal of SR1 MOSFET per Figure 125

Ch1: V_{OUT}, 50 mV/div

Ch4: I_{OUT}, 10 A/div

Ch2: Q1 PWM, 2 V/div, PWM control signal of Q1 MOSFET per Figure 125

Figure 126 Burst mode entry criteria, load changed from 10 A to 4 A, at 48 V input

9.2.2 Burst mode exit criteria

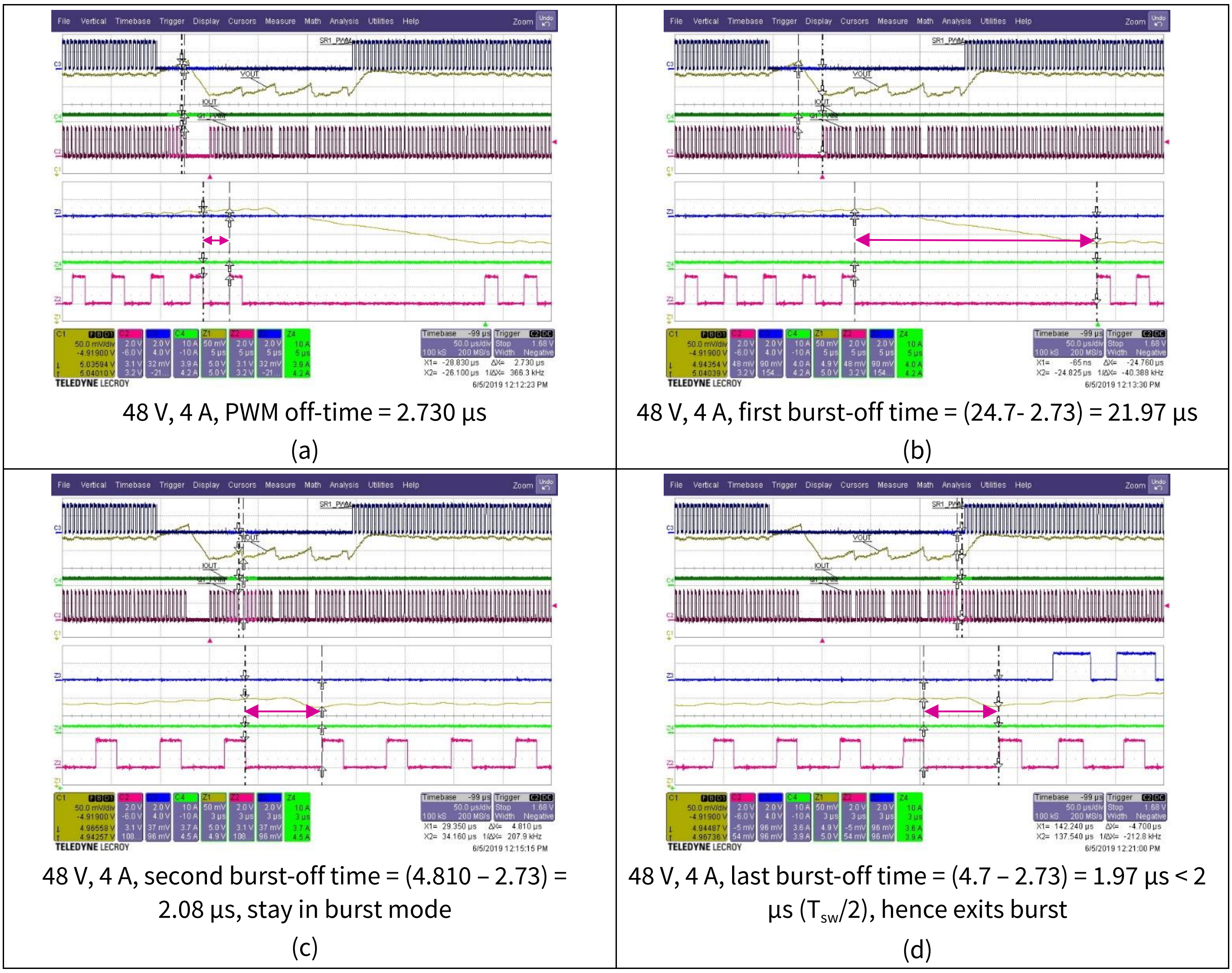
The exit criteria for burst mode are that the burst-off time should be less than half of the switching cycle time. The burst-off time is calculated by the off-time between two burst-ons minus the PWM off-time. The **pid0_burst_reps** was set to 3 in this example. The switching frequency of the converter was 250 kHz, i.e. T_{SW} is 4 µs. The converter will exit burst mode when the burst-off time is less than half of T_{SW}, i.e. 2 µs.

Channel 1 (yellow) – V_{OUT}, 50 mV/div, 10 µs/div – represents the converter output voltage.

Channel 2 (red) – Q1 PWM, 2 V/div, 10 µs/div – represents PWM control signal of Q1 MOSFET per **Figure 125**.

Channel 3 (blue) – SR1 PWM, 2 V/div, 10 µs/div – represents PWM control signal of SR1 MOSFET per **Figure 125**.

Channel 4 (green) – I_{OUT}, 10 A/div, 10 µs/div – represents the output load current.



48 V, 4 A, PWM off-time = 2.730 µs
(a)

48 V, 4 A, first burst-off time = (24.7- 2.73) = 21.97 µs
(b)

48 V, 4 A, second burst-off time = (4.810 – 2.73) = 2.08 µs, stay in burst mode
(c)

48 V, 4 A, last burst-off time = (4.7 – 2.73) = 1.97 µs < 2 µs (T_{sw}/2), hence exits burst
(d)

Figure 127 Burst mode exit criteria

The bottom half of each waveform is the detail of the highlighted portion of the top waveform. Waveform (a) measured the PWM off-time in a normal switching cycle. This number is used to calculate the burst-off time. Waveforms (b), (c) and (d) measured the off-time between two burst-ons in a different section. In waveform (d) the burst-off time is shorter than 2 µs and the IC exited burst at this point.

9.3 Burst mode reps

The XDPP1100 controller enables the user to configure the number of pulses in burst mode using register **pidX_burst_reps**. The range of values allowed are 0 to 3, and the number of pulses = 2^(value in pidX_burst_reps). Here the X is 0 or 1, referring to the register of loop 0 or loop 1.

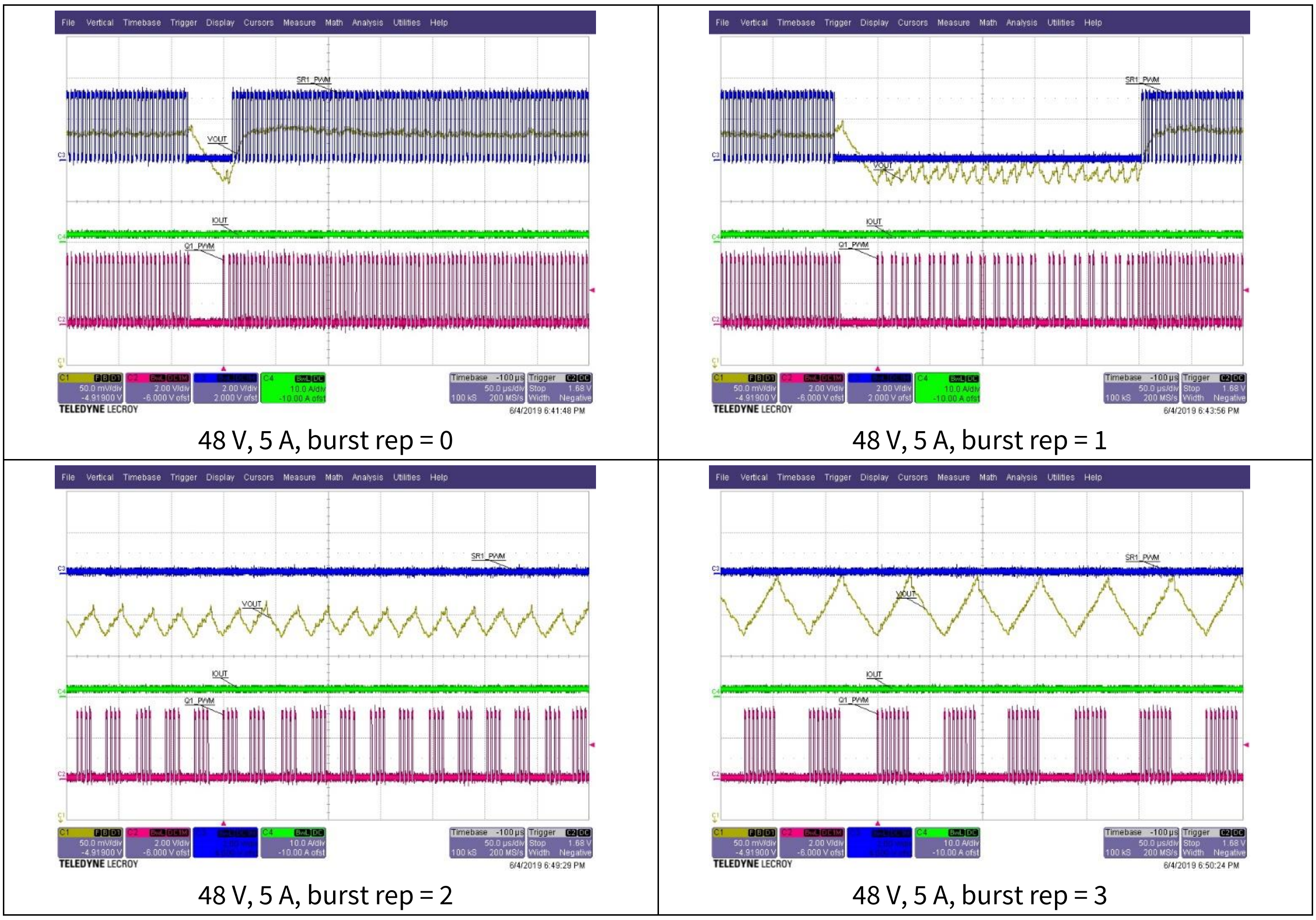
The description of channels used in **Figure 128** is as follows:

Channel 1 (yellow) – V_{OUT}, 50 mV/div, 10 µs/div – represents the converter output voltage.

Channel 2 (red) – Q1 PWM, 2 V/div, 10 µs/div – represents PWM control signal of Q1 MOSFET per **Figure 125**.

Channel 3 (blue) – SR1 PWM, 2 V/div, 10 µs/div – represents PWM control signal of SR1 MOSFET per **Figure 125**.

Channel 4 (green) – I_{OUT}, 10 A/div, 10 µs/div – represents the output load current.



48 V, 5 A, burst rep = 0

48 V, 5 A, burst rep = 1

48 V, 5 A, burst rep = 2

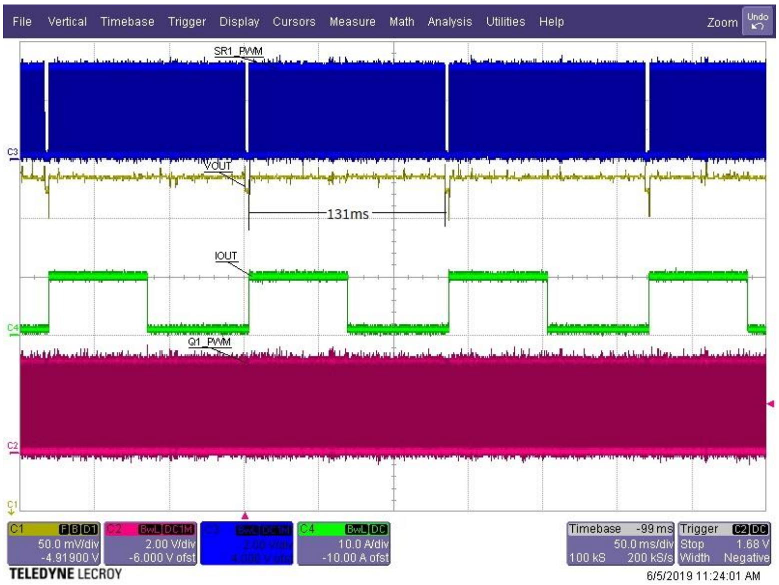
48 V, 5 A, burst rep = 3

Figure 128 Burst mode reps

A low burst rep (i.e. burst rep = 0) could cause immediate exit from burst after an entry, which would result in no power-saving effect with the burst operation. A larger burst rep will result in larger V_{OUT} ripple during burst. The configuration should be optimized based on application requirements, i.e. it depends on what load level is needed to enter and stay in burst mode. When the burst entry threshold is low, the number of burst reps could also be set low to maintain low output ripple.

9.4 Burst re-entry counter

The XDPP1100 controller has a hold-off re-entry timer for 2^15 x T_{sw} in burst mode, i.e. once the system exits burst mode it will only re-enter it after the specified time. This function prevents the immediate re-entry of burst mode after an exit. This is referred to as “burst re-entry timer”. The waveform shown in **Figure 129** explains this concept with a 250 kHz switching frequency. The burst hold-off time is 2^15*4 µs = 131 ms.



Ch3: SR1 PWM, 2 V/div, PWM control signal of SR1 MOSFET per Figure 125

Ch1: V_{OUT}, 50 mV/div

Ch4: I_{OUT}, 10 A/div

Ch2: Q1 PWM, 2 V/div, PWM control signal of Q1 MOSFET per Figure 125

Figure 129 Burst re-entry counter

Figure 129 is the HBCT converter tested at 48 V input, at dynamic load changing between 10 A and 1 A. The 1A load duration was 66 ms, and 10 A load duration was 68 ms. The system did not enter burst mode immediately when current reduced from 10 A to 1 A because of the hold-off time of 131 ms. It entered burst only for the last 3 ms, i.e. (66+68) ms - 131 ms = 3 ms.

9.5 Load transient during burst mode

A load transient was performed during burst mode operation from 0 A to 10 A to study the impact on the output voltage droop.

9.5.1 Load transient with linear PID

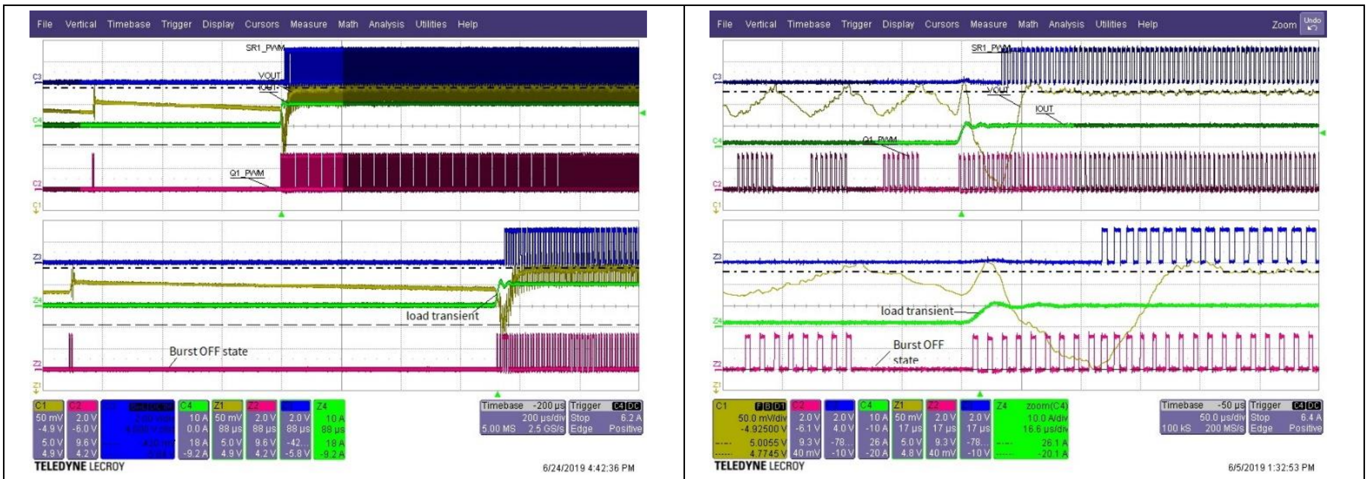
The burst_reps register is set to 0 and 3 in this test. The linear loop PID responds to the load transient by exiting burst mode. The V_{OUT} undershoot observed was 135.5 mV with one switching cycle (burst rep = 0) and 231 mV with eight switching cycles (burst rep = 3). The description of channels used in **Figure 130** is as mentioned below. Please note that a higher number of burst_reps would result in a longer wait to exit burst and therefore a larger undershoot.

Channel 1 (yellow) – V_{OUT}, 50 mV/div, 10 µs/div – represents the converter output voltage.

Channel 2 (red) – Q1 PWM, 2 V/div, 10 µs/div – represents PWM control signal of Q1 MOSFET per **Figure 125**.

Channel 3 (blue) – SR1 PWM, 2 V/div, 10 µs/div – represents PWM control signal of SR1 MOSFET per **Figure 125**.

Channel 4 (green) – I_{OUT}, 10 A/div, 10 µs/div – represents the output load current.



48 V, 0 to 10 A, burst rep = 1, V_{OUT} undershoot 135.5 mV

48 V, 2 to 10 A, burst rep = 3, V_{OUT} undershoot 231 mV

Figure 130 Load transition during burst off mode

9.5.2 Load transient with FTR enabled

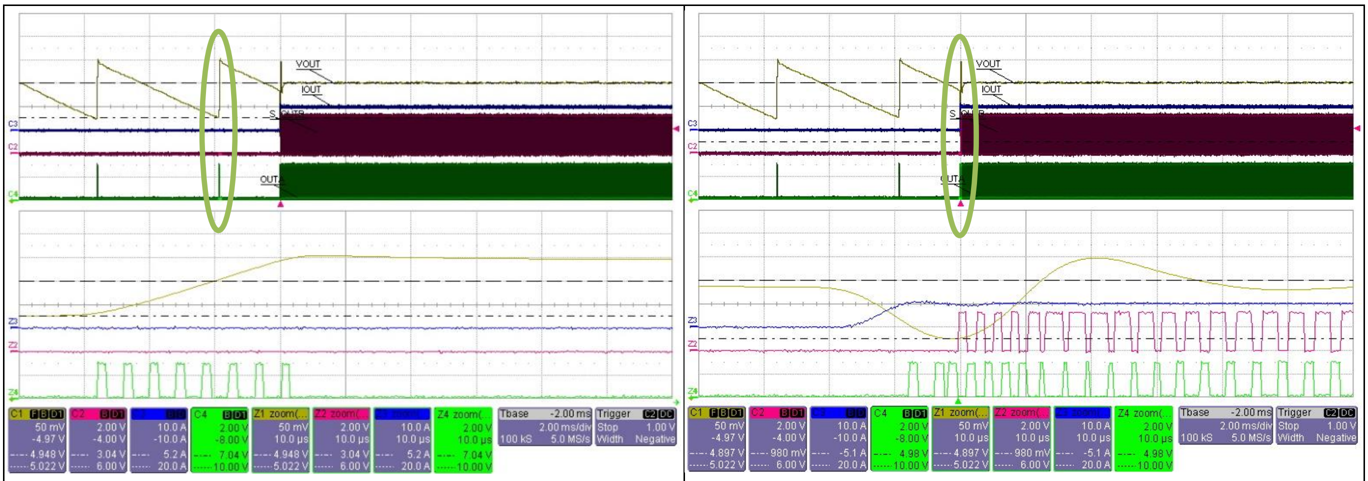
The XDPP1100 provides a FTR feature to reduce undershoot or overshoot during load transient. If the FTR is enabled, the V_{OUT} undershoot in load transient during burst mode can be reduced. When the load transient occurs during burst mode, the FTR would take over PWM duty-cycle control immediately without waiting for the completion of the burst rep cycles. In this test, the burst_reps register is set to 3 for eight switching cycles during the burst-on phase.

Channel 1 (yellow) – V_{OUT}, 50 mV/div, 10 µs/div – represents the converter output voltage.

Channel 2 (red) – SR1 PWM, 2 V/div, 10 µs/div – represents PWM control signal of SR1 MOSFET per **Figure 125**.

Channel 3 (blue) – I_{OUT}, 10 A/div, 10 µs/div – represents the output load current.

Channel 4 (green) – Q1 PWM, 2 V/div, 10 µs/div – represents PWM control signal of Q1 MOSFET per **Figure 125**.



There were eight switching cycles during a normal burst-on.

(a)

The FTR took over control during the burst-on phase on detection of load transient.

(b)

Figure 131 Load transition during burst with FTR enabled

With waveform (b), Q1 switched two cycles and exited burst. The switching frequency increased to $1/(2*T_{on})$ for 100 percent duty-cycle and the SR was enabled. V_{OUT} undershoot reduced from 230 mV to 100 mV during the load transient.

9.6 Power loss reduction with burst mode

The power loss reduction in burst mode using HBCT is shown in Table 67. It is evident that the input current consumption with burst mode at a given load current is lower than in regular mode. At 48 V input and 1 A load, the power saving is 1.1 W with burst mode and burst rep = 3.

Table 67 Burst mode power loss reduction

Input voltage (V)	Input current (A)	Input power (W)	Output voltage (V)	Output current (A)	Burst mode	Burst reps
48	0.2702	12.97	5.002	2	On	0
48	0.2702	12.97	5.002	2	On	1
48	0.2592	12.44	4.961	2	On	2
48	0.2581	12.39	4.958	2	On	3
48	0.1986	9.53	4.985	1.5	On	3
48	0.2156	10.35	5.002	1.5	Off	NA
48	0.1368	6.57	4.990	1	On	3
48	0.16	7.68	5.002	1	Off	NA

9.7 Summary of burst operation

The XDPP1100 provides flexibility for the user to configure burst operation according to application requirements. With burst mode operation, light load power losses are reduced. Burst mode operation is recommended when the light load power consumption is more critical than the load transient performance.

10 Synchronization

In power supply design, synchronizing multiple converters to the same frequency is often used to reduce the filter cost, lower the EMI and reduce the voltage and current ripple with proper phase shift. This chapter describes the procedure to configure the sync in/out feature of the XDPP1100 digital controller.

The sync in/out feature of the XDPP1100 allows customers to synchronize the PWM outputs to an external signal or provide a sync signal for other converters to act in synchronization. Some of the applications include interleaved topologies, parallel converters with phase shift, etc. The sync pin can be configured as either an input or an output. The maximum range of sync in signal is ±6.25 percent of the programmed switching frequency. Once the frequency is locked, the system can remain in sync for a maximum of ±12.5 percent of the programmed switching frequency. The XDPP1100 sync out signal is a square waveform and is synchronized to PWM ramp 0. The phase of the sync out signal depends on converter modulation type. Any GPIO-capable pin can be mapped to perform the sync function by setting the pin function register to 3. This document uses the SYNC pin to demonstrate the XDPP1100 sync in/out feature.

10.1 Configuring sync in/out

The sync-in function can be configured to sync with either an internal signal or an external clock. The sync to internal clock can be categorized into four different applications: single-phase; single-loop (loop 0) dual-phase (interleaved); dual-loop same switching frequency; dual-loop different switching frequency. External sync is possible in all the cases except for dual-loop with different switching frequency.

10.1.1 Sync-in configuration using internal signal

Table 68 explains the configuration procedure for using an internal sync clock with description of the registers.

Table 68 Register settings for sync to internal ramp based on application

Application	Register name	Description
Single-phase	ramp0_sync_sel	Sync select for ramp 0. Ramp 0 is used on **single-phase (loop 0)**, interleaved (phase 0) or dual-loop (loop 0) designs. = 1, should always be set to 1
	ramp1_sync_sel	Sync select for ramp 1. Ramp 1 is used on interleaved (phase 1) or dual-loop (loop 1) designs. = 1 or 0, does not matter, ramp 1 not used
Interleaved	ramp0_sync_sel	Sync select for ramp 0. Ramp 0 is used on single-phase (loop 0), **interleaved (phase 0)** or dual-loop (loop 0) designs. = 1, should always be set to 1
	ramp1_sync_sel	Sync select for ramp 1. Ramp 1 is used on **interleaved (phase 1)** or dual-loop (loop 1) designs. = 1, **sync to Fswitch0** or external sync
Dual-loop with same switching frequency	ramp0_sync_sel	Sync select for ramp 0. Ramp 0 is used on single-phase (loop 0), interleaved (phase 0) or **dual-loop (loop 0)** designs. = 1, should always be set to 1
	ramp1_sync_sel	Sync select for ramp 1. Ramp 1 is used on interleaved (phase 1) or **dual-loop (loop 1)** designs. = 1, **sync to Fswitch0** or external sync
Dual-loop with different	ramp0_sync_sel	Sync select for ramp 0. Ramp 0 is used on single-phase (loop

Application	Register name	Description
switching frequency		0), interleaved (phase 0) or **dual-loop (loop 0)** designs. = 1, should always be set to 1
	ramp1_sync_sel	Sync select for ramp 1. Ramp 1 is used on interleaved (phase 1) or **dual-loop (loop 1)** designs. = 0, **sync to Fswitch1**/no external sync

10.1.2 Sync-in configuration using external signal

Table 69 explains the configuration procedure for using an external sync signal with description of the registers.

Table 69 Register settings for sync to external signal

Register name	Description
Sync_func	Pin SYNC function definition. = 3, configure the function of SYNC pin as sync
sync_dir_out	Defines direction of pin mapped to SYNC function = 0, configure sync function as an input
sync_deglitch_en	De-glitch enable for digital sync function when used as an input. = 1, enable sync de-glitch to avoid ramp_sync response to noise/spikes Sync de-glitch filter add 40 ns delay.
ramp0_sync_sel	Sync select for ramp 0. Ramp 0 is used on single-phase (loop 0), interleaved (phase 0) or dual-loop (loop 0) designs. = 1, should always be set to 1
ramp1_sync_sel	Sync select for ramp 1. Ramp 1 is used on interleaved (phase 1) or dual-loop (loop 1) designs. = 1, sync to external signal Set switching frequency of loop 1 the same as that of loop 0 to allow synchronization.
ramp0_m_flavor	Modulation type for ramp 0. 0 = Dual edge, 1 = Leading edge, 2 = Trailing edge.
ramp1_m_flavor	Modulation type for ramp 1. 0 = Dual edge, 1 = Leading edge, 2 = Trailing edge
sync_pd	Pin SYNC weak pull-down enable. = 1, weak pull-down is enabled
sync_ppen	Pin SYNC output buffer CMOS/open drain select. = 0, open drain Since SYNC is an input pin in this configuration, open drain output is recommended.
sync_pu_n	Pin SYNC weak pull-up enable. = 1, weak pull-up is disabled

10.1.3 Sync-out configuration

Table 70 explains the sync-out configuration procedure with a clear description of the registers.

Table 70 Register settings for sync-out signal

Register	Description
Sync_func	Pin SYNC function definition. = 3, configure the function of SYNC pin as sync
sync_dir_out	Defines direction of pin mapped to sync function. = 1, configure sync function as an output
sync_ppen	Pin SYNC output buffer CMOS/open drain select. = 1, CMOS output = 0, open drain output, requires external pull-up resistor If the application has an external pull-up resistor to the SYNC pin, it does not matter if sync_ppen is 0 or 1.
ramp0_m_flavor	Modulation type for ramp 0. 0 = Dual edge, 1 = Leading edge, 2 = Trailing edge. Sync-out signal alignment varies with modulation type. Sync-out is possible only for the ramp 0 signal.
sync_pd	Pin SYNC weak pull-down enable. = 0, weak pull-down is disabled
sync_pu_n	Pin SYNC weak pull-up enable. = 1, weak pull-up is disabled

10.1.4 Sync-in/out registers

Table 71 provides a summary of registers used for sync-in/out configuration.

Table 71 Sync-in/out registers

Name	Address	Bits	Description
Common peripheral			
sync_func	7000_3008_H	[20:18]	Pin SYNC function definition. 0 not used 1 IO, GPIO0 [7] 2 IO, GPIO1 [7] 3 IO, SYNC 4 output, FAN1_PWM 5 to 7, not used
sync_pd	7000_3008_H	[21]	Pin SYNC weak pull-down enable. = 0, pull-down is disabled = 1, pull-down is enabled
sync_pu_n	7000_3008_H	[22]	Pin SYNC weak pull-up enable. = 0, pull-up is enabled = 1, pull-up is disabled
sync_ppen	7000_3008_H	[23:23]	Pin SYNC output buffer CMOS/open drain select. 0 “open drain output” 1 “CMOS output”

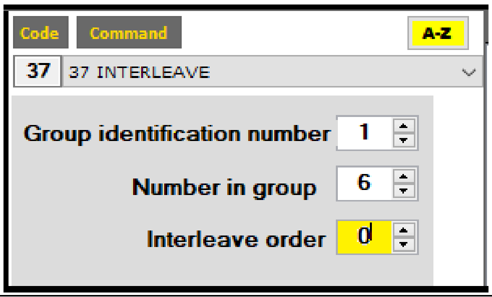
Name	Address	Bits	Description
sync_dir_out	7000_3014_H	[13:13]	Defines direction of pin mapped to SYNC function. 0 “SYNC mapped pin is input” 1 “SYNC mapped pin is output”
sync_de-glitch_en	7000_3008_H	[28:28]	De-glitch enable for digital sync function when used as an input. 0 “Sync de-glitch disabled” 1 “Sync de-glitch enabled”
sync_state	7000_30A8_H	[23:22]	Digital SYNC state. This is a read-only register, indicates the state of sync. 0 “using internal sync clock” 1 “phase locking to external sync clock” 2 “using external sync clock” 3 “phase locking to internal sync clock”
PWM peripheral			
ramp_sync_sel	7000_2C00_H 7000_2C00_H	[1] [7]	Bit [1], sync select for ramp 0. Should always be set to 1. Bit [7], sync select for ramp 1. ramp1_sync_sel “sync to” cases: 0 “Fswitch1/no external sync”, “Dual-loop with Fsw1 = Fsw0, loop 1 only” 1 “Fswitch0 or external sync”, “Dual-loop with Fsw1 = Fsw0, single-loop interleave phase”
ramp_m_flavor	7000_2C00_H	[3:2] [9:8]	Modulation type for ramp 0/1 00 “Dual edge” 01 “Leading edge” 1x “Trailing edge”

10.1.5 Sync-in/out PMBus commands

Below is the only PMBus command required for sync-in/out. The INTERLEAVE command is used to configure the phase of interleaved topologies or the phase of multiple devices in parallel unit applications.

Table 72 Sync-in/out PMBus command

Command name	Command code	Description
INTERLEAVE	37_H	The INTERLEAVE command is used to arrange multiple units so that their switching periods can be distributed in time. This may be used to facilitate paralleling of multiple units or to reduce AC currents injected into the power bus. The INTERLEAVE command is also used to configure the phase of each ramp of the interleaved topologies.



37 | 37 INTERLEAVE

Group identification number: 1

Number in group: 6

Interleave order: 0

Figure 132 INTERLEAVE command

The phase shift (lag) is defined by: 360° × $\frac{Interleave\_order}{Number\_in\_group}$

10.2 Different types of PWM

The XDPP1100 controller can be configured for different types of PWM such as trailing-edge, leading-edge, and dual-edge modulation. **Figure 133** demonstrates the difference between different PWM modulation types. The trailing-edge modulation has PWM rising edge aligned to the beginning of the ramp, the leading-edge modulation has PWM falling edge aligned to the end of the ramp, and the dual-edge modulation has PWM positioned at the middle of the ramp. Please note that the VMC supports all three types of modulation and the PCMC must use trailing-edge modulation.

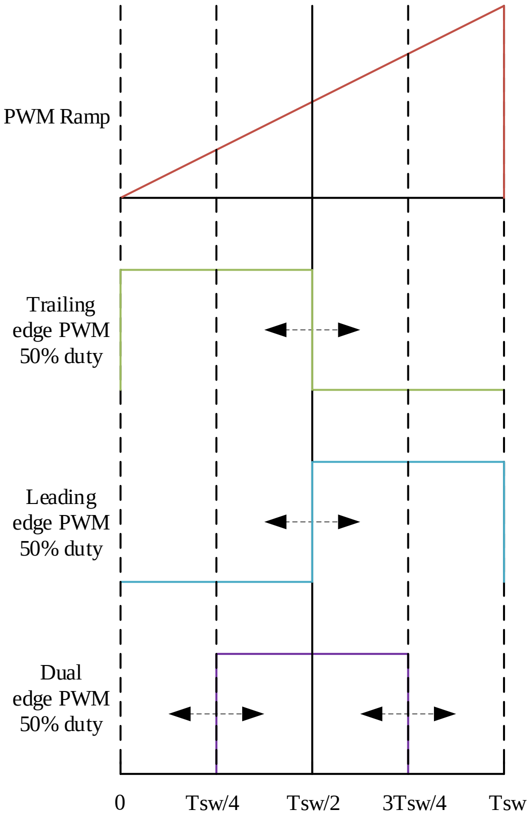


Figure 133 PWM modulation types

A test was performed using HBCT topology as shown in **Figure 134** to demonstrate different PWM modulation techniques using the XDPP1100 controller.

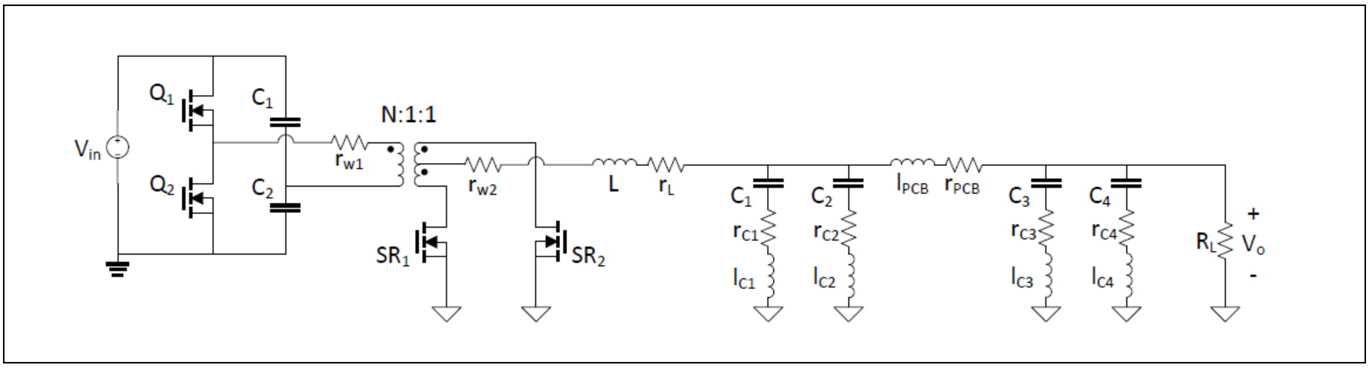


Figure 134 HBCT schematic

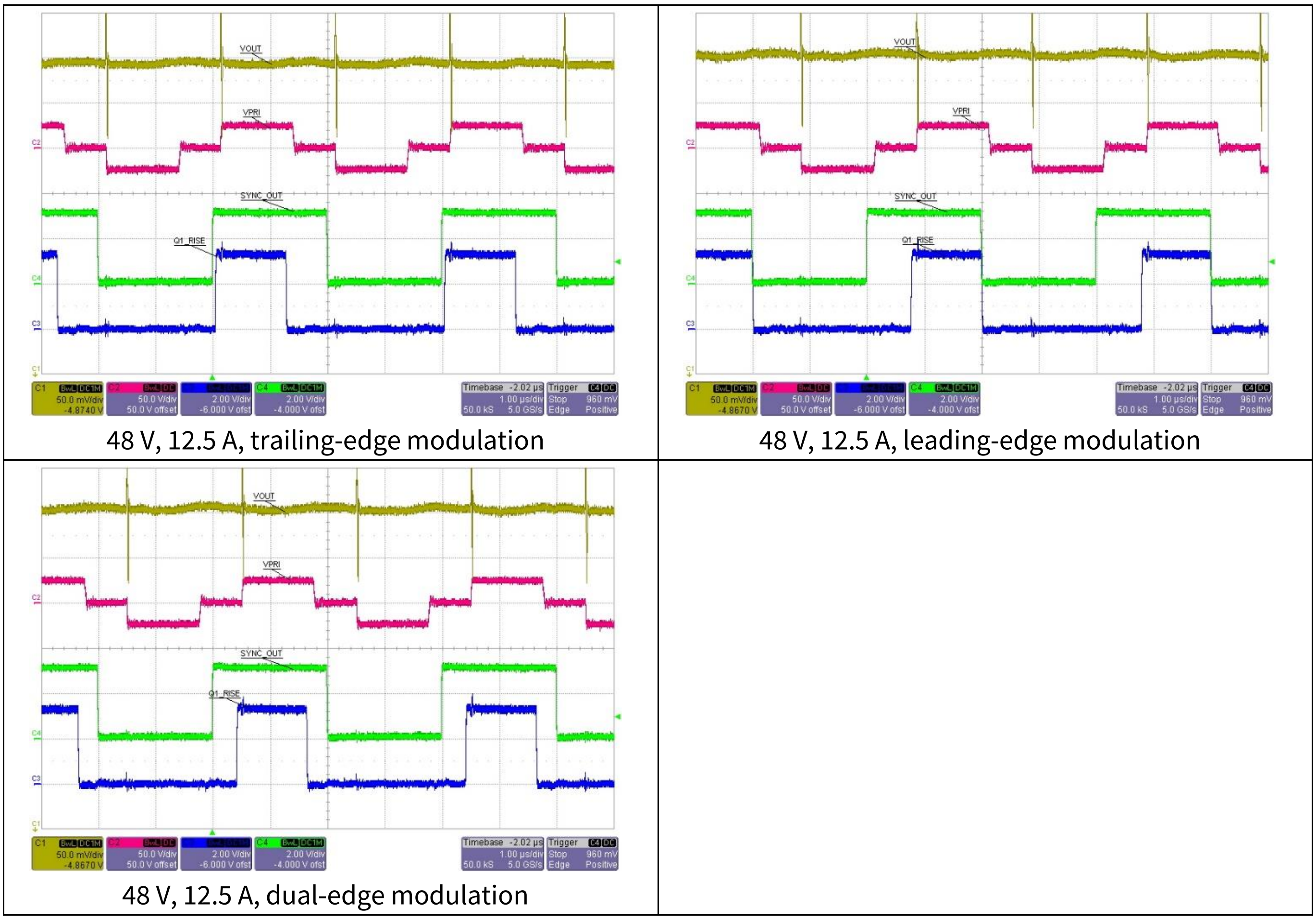
The description of channels used in **Figure 135** is as follows:

Channel 1 (yellow) – V_{OUT}, 50 mV/div, 1 µs/div – represents the converter output voltage.

Channel 2 (red) – V_{pri}, 50 V/div, 1 µs/div – represents voltage across transformer primary.

Channel 3 (blue) – Q1 PWM, 2 V/div, 1 µs/div – represents PWM control signal of Q1 MOSFET per **Figure 134**.

Channel 4 (green) – sync out, 2 V/div, 1 µs/div – represents the internal sync signal brought out using SYNC pin.



48 V, 12.5 A, trailing-edge modulation

48 V, 12.5 A, leading-edge modulation

48 V, 12.5 A, dual-edge modulation

Figure 135 Waveforms of different PWM types

10.3 Sync-in for different applications

The sync-in for different modulation schemes can be identified by referring to Table 73. **Figure 134** is an example of bridge topology. **Figure 140** is an example of ACF topology.

Table 73 Sync rising edge locations

Topology	Trailing edge	Leading edge	Dual edge
Bridge	At Q1 rise/SR2 fall	At Q2 fall/SR1 rise	Between Q2 fall and Q1 rise
ACF (P-channel clamp)	At CLAMP rise	At CONTROL fall	Between CONTROL fall and rise

10.3.1 Using external sync-in signal

This test was performed using an external function generator. The function generator was used to create a square waveform with amplitude 0 V to 3.3 V, 50 percent duty-cycle, as an external sync signal. This setup demonstrates applications involving multiple XDPP1100 controllers for proper synchronization during paralleling applications. Table 73 is used as reference for determining the position of the PWMs with reference to the sync-in signal. The waveforms for different types of modulation are shown below, captured using HBCT topology per **Figure 134**.

For uniformity, a common configuration was used in PMBus command 37, i.e. group identification number = 1, number in group = 6, interleave order = 0 to 5 for all the waveforms captured. The interleave phase for different interleave orders can be determined by the following equation:

$$\text{Interleave phase} = \frac{Interleave\ order}{Number\ in\ group} \times 360°, \text{lagging}$$

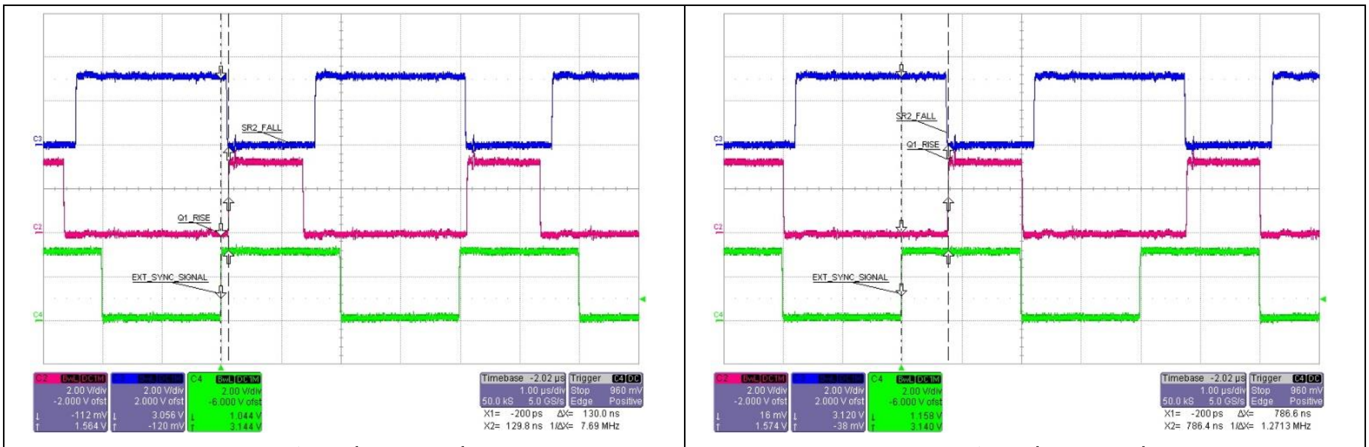
10.3.1.1 Trailing edge modulation

This section demonstrates waveforms using trailing edge modulation. Sync frequency is 250 kHz.

Channel 2 (red) – Q1 rise, 2 V/div, 1 µs/div – represents PWM control signal of Q1 FET per **Figure 134**.

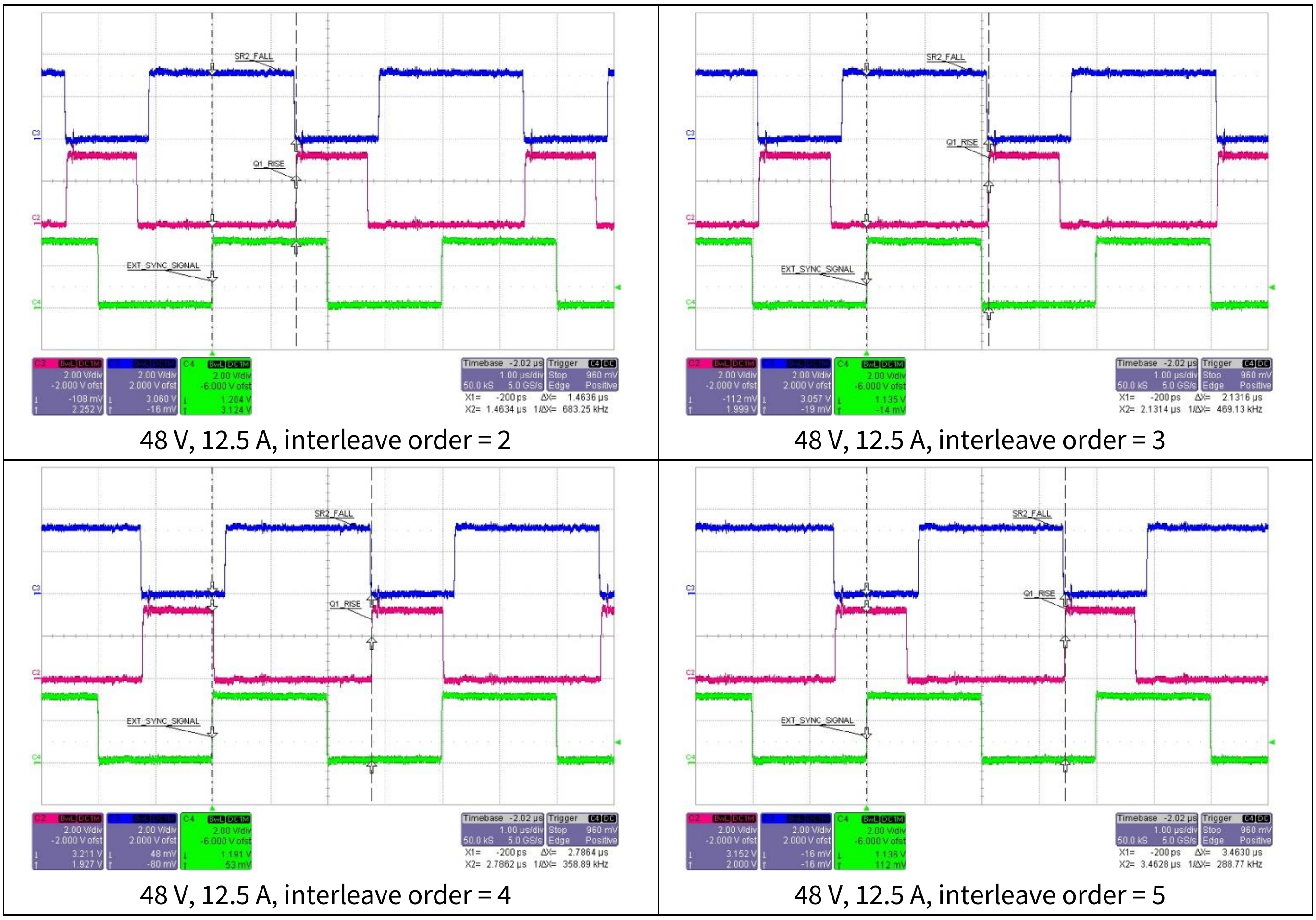
Channel 3 (blue) – SR2 fall, 2 V/div, 1 µs/div – represents PWM control signal of SR2 FET per **Figure 134**.

Channel 4 (green) – external sync signal, 2 V/div, 1 µs/div – represents function generator signal.



48 V, 12.5 A, interleave order = 0

48 V, 12.5 A, interleave order = 1



48 V, 12.5 A, interleave order = 2

48 V, 12.5 A, interleave order = 3

48 V, 12.5 A, interleave order = 4

48 V, 12.5 A, interleave order = 5

Figure 136 **Interleave trailing edge waveforms**

When the interleave order is 0, the delay from the rising edge of the external sync signal to the rising edge of Q1 is the sum of the following items. It is 130 ns in this test example:

- Sync de-glitch filter, 40 ns (if enabled)
- Sync block propagation delay, 30 ns
- Q1 rising-edge dead-time, user configurable

When the interleave order is non-zero, the delay from the rising edge of the external sync signal to the rising edge of Q1 is the 0 phase delay plus the interleaved phase-shift. For example, when the interleave order = 4, the Q1 rising edge is 240 degrees, lagging to the rising edge of the sync-in pulse. The switching frequency is 250 kHz, the phase shift is 2.67 µs. Total phase lagging is 0.13 µs + 2.67 µs = 2.8 µs.

$$\text{Interleave phase} = \frac{Interleave\ order}{Number\ in\ group} \times 360° = \frac{4}{6} \times 360° = 240°$$

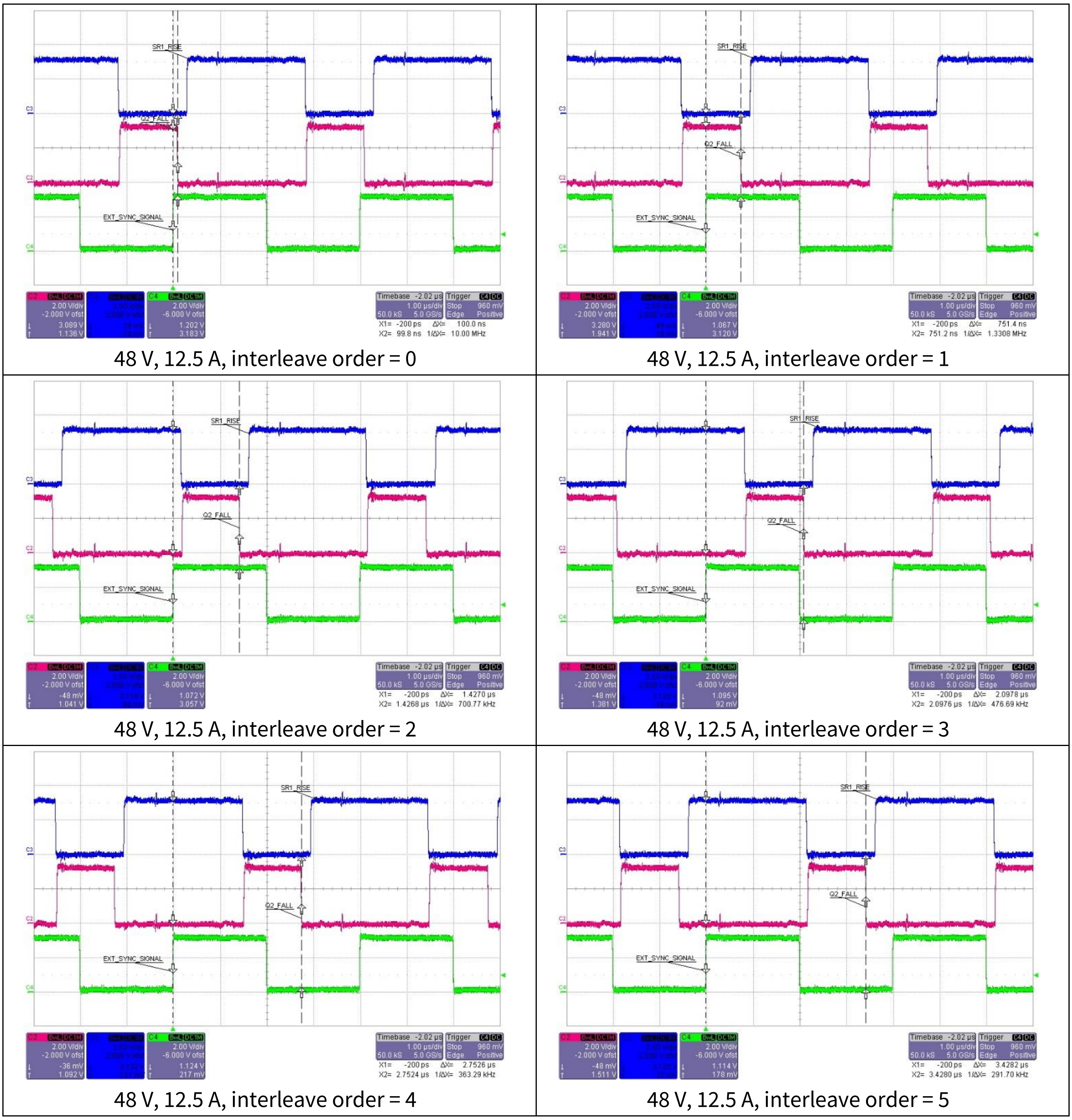
10.3.1.2 Leading edge modulation

This section demonstrates waveforms using leading edge modulation.

Channel 2 (red) – Q2 fall, 2 V/div, 1 µs/div – represents PWM control signal of Q2 MOSFET per **Figure 134**.

Channel 3 (blue) – SR1 rise, 2 V/div, 1 µs/div – represents PWM control signal of SR1 MOSFET per **Figure 134**.

Channel 4 (green) – external sync signal, 2 V/div, 1 µs/div – represents function generator signal.



48 V, 12.5 A, interleave order = 0

48 V, 12.5 A, interleave order = 1

48 V, 12.5 A, interleave order = 2

48 V, 12.5 A, interleave order = 3

48 V, 12.5 A, interleave order = 4

48 V, 12.5 A, interleave order = 5

Figure 137 Interleave leading edge waveforms

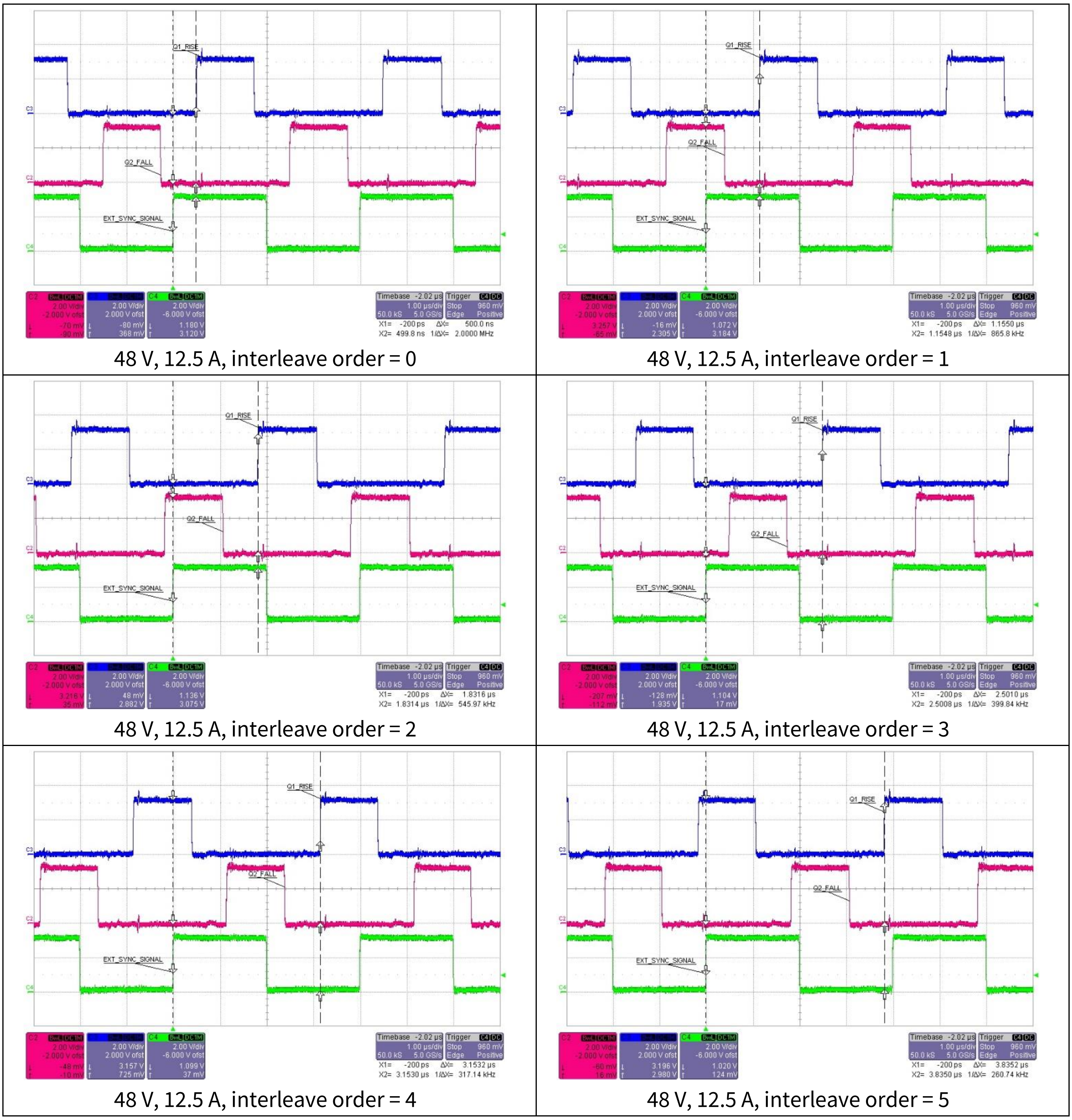
10.3.1.3 Dual edge modulation

This section demonstrates waveforms using dual edge modulation.

Channel 2 (red) – Q2 fall, 2 V/div, 1 µs/div – represents PWM control signal of Q2 MOSFET per **Figure 134**.

Channel 3 (blue) – Q1 rise, 2 V/div, 1 µs/div – represents PWM control signal of Q1 MOSFET per **Figure 134**.

Channel 4 (green) – external sync signal, 2 V/div, 1 µs/div – represents function generator signal.



48 V, 12.5 A, interleave order = 0

48 V, 12.5 A, interleave order = 1

48 V, 12.5 A, interleave order = 2

48 V, 12.5 A, interleave order = 3

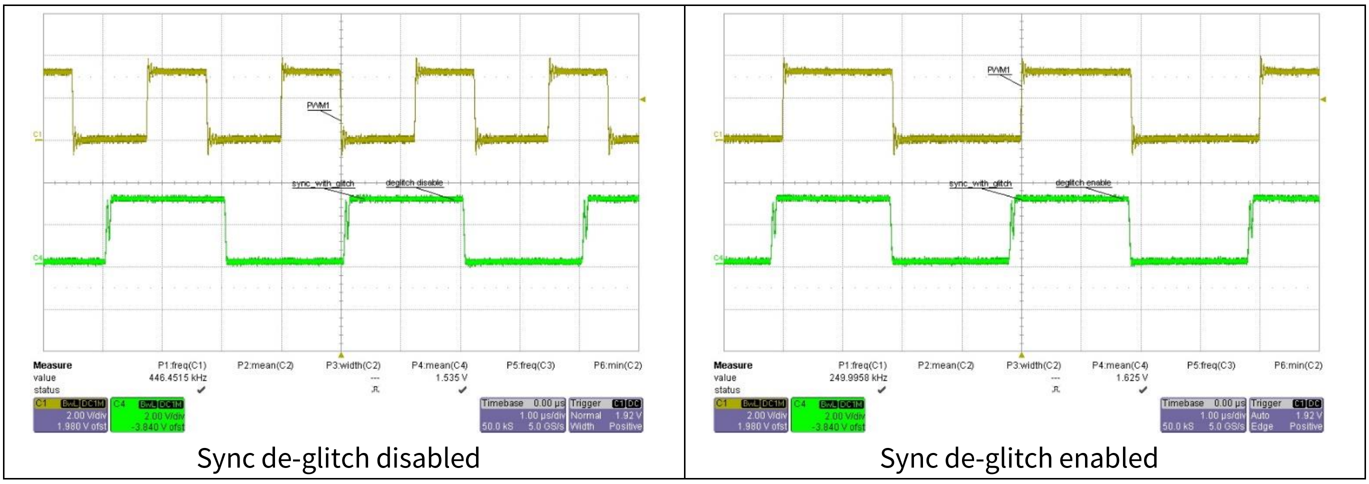
48 V, 12.5 A, interleave order = 4

48 V, 12.5 A, interleave order = 5

Figure 138 Interleave dual-edge waveforms

10.3.1.4 Impact of sync-in de-glitch

This section demonstrates the importance of sync-in de-glitch in avoiding the noise/spikes in the external sync signal. This test was performed by creating a high-frequency noise in the function generator sync signal.



Sync de-glitch disabled

Sync de-glitch enabled

Figure 139 Sync de-glitch waveforms

Channel 1 (yellow) – PWM control signal, 2 V/div, 1 µs/div.

Channel 4 (green) – sync signal with a glitch, 2 V/div, 1 µs/div.

It can be observed from the above waveform that the PWM signal is asynchronous with respect to the sync signal with a glitch when sync de-glitch is disabled. The sync de-glitch filter adds 40 ns delay to the sync function. It is recommended to enable the de-glitch filter only if the sync signal is noisy.

10.3.2 Sync-in internal

This section describes the configuration of the sync-in signal for a single XDPP1100 controller for dual-loop or interleaved application. The dual-loop control can only be achieved using a performance version XDPP1100-Q040. The waveforms are captured as in various cases mentioned in section 10.1.1. In order to capture waveforms, the internal sync clock has been brought out using a sync-out signal. Trailing edge modulation was used for this test.

10.3.2.1 Loop 0, single phase

Please refer to section 10.1.1 for configuration. A single-phase ACF topology as shown in **Figure 140** was used to capture waveforms for this test.

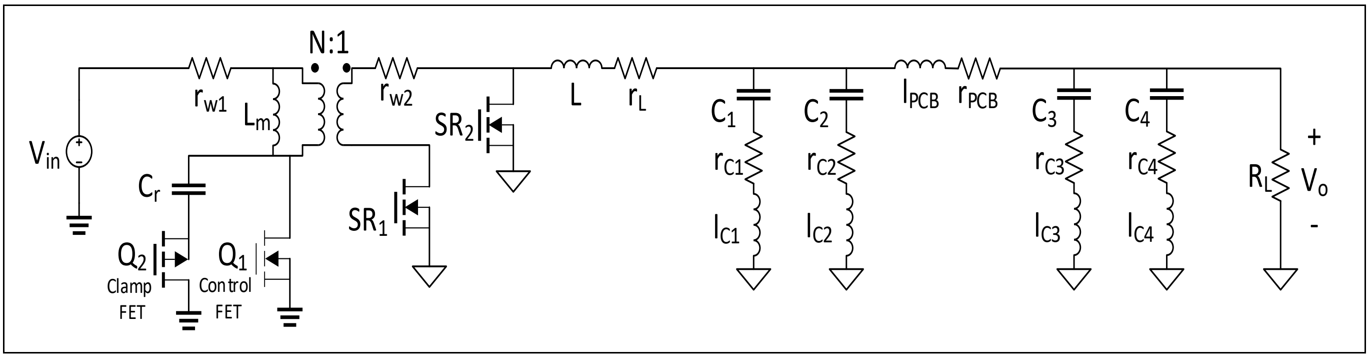


Figure 140 Single-phase ACF schematic

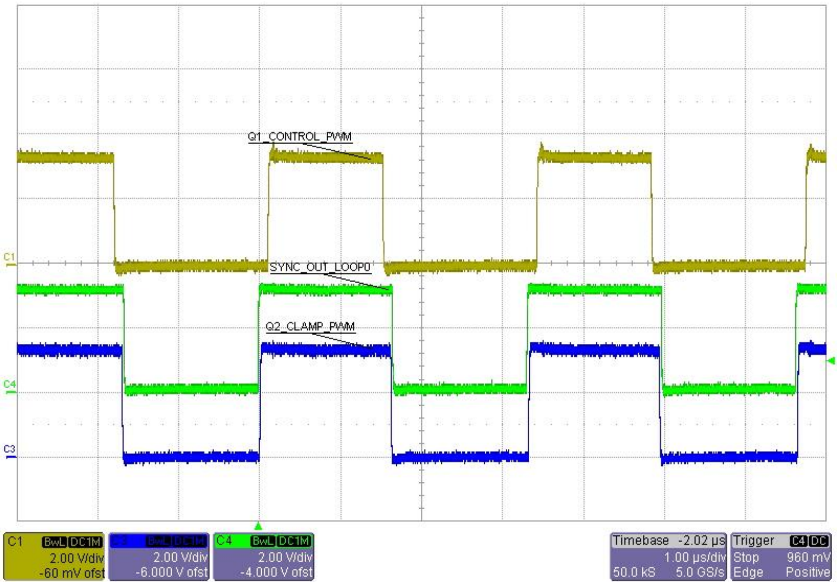

Q1_CONTROL_PWM
SYNC_OUT_LOOP0
Q2_CLAMP_PWM
C1
C4
C3
C1 BwL DC1M 2.00 V/div -60 mV ofst
C3 BwL DC1M 2.00 V/div -6.000 V ofst
C4 BwL DC1M 2.00 V/div -4.000 V ofst
Timebase -2.02 µs 1.00 µs/div 50.0 kS 5.0 GS/s
Trigger C4 DC Stop 960 mV Edge Positive

Figure 141 Single-phase ACF waveforms

Channel 1 (yellow) – Q1_Control_PWM, 2 V/div, 1 µs/div – represents PWM control signal of Q1 MOSFET per **Figure 140**.

Channel 2 (blue) – Q2_Clamp_FET, 2 V/div, 1 µs/div – represents PWM control signal of Q2 MOSFET per **Figure 140**.

Channel 4 (green) – Sync_Out_Loop0, 2 V/div, 1 µs/div – represents internal sync clock.

10.3.2.2 Loop 0, dual-phase interleaved

Please refer to section 10.1.1 for configuration. In addition to this, the PMBus interleave command should be set as group identification number = 1, number in group = 2, interleave order = 0 (loop 0), interleave order = 1 (loop 1). A dual-phase ACF topology as shown in **Figure 142** was used to capture waveforms for this test.

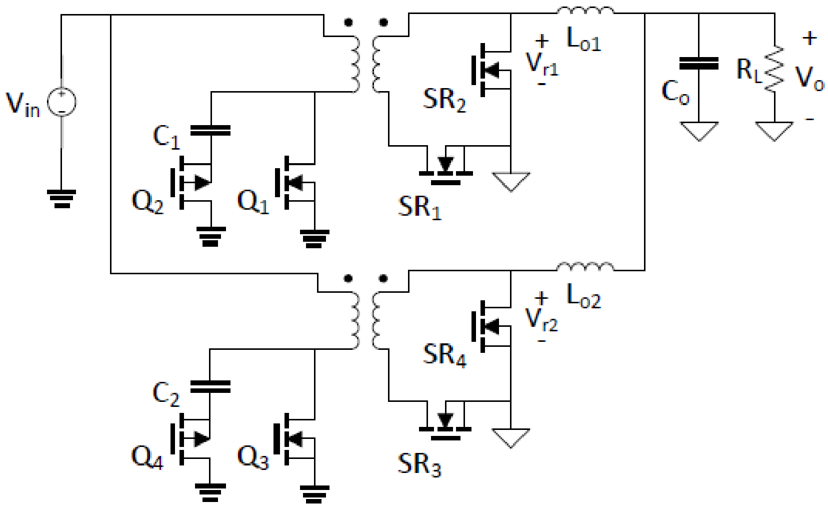


Figure 142 Interleaved ACF schematic

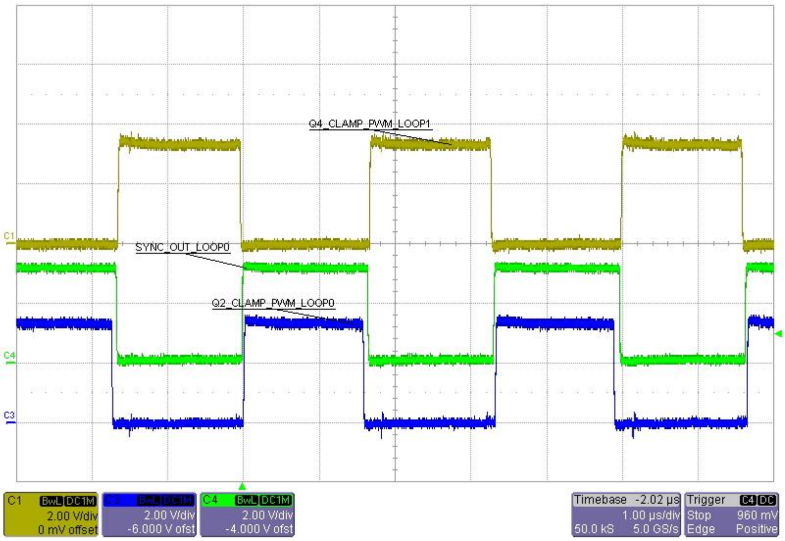


Figure 143 Dual-phase interleaved waveforms

Channel 1 (yellow) – Q4_Clamp_PWM (loop1), 2 V/div, 1 µs/div – represents PWM control signal of Q4 MOSFET per **Figure 142**.

Channel 2 (blue) – Q2_Clamp_FET (loop0), 2 V/div, 1 µs/div – represents PWM control signal of Q2 MOSFET per **Figure 142**.

Channel 4 (green) – Sync_Out_Loop0, 2 V/div, 1 µs/div – represents internal sync clock.

10.3.2.3 Dual-loop

Please refer to section 10.1.1 for configuration. A dual-phase ACF topology as shown in **Figure 144** was used to capture waveforms for this test. Dual-loop application refers to a condition of using a single XDPP1100-Q040 controller for control of two different loops with a dual output. These can be configured to operate with an interleaved control signal or without interleave, and it can also be categorized as having the same switching frequency or a different switching frequency. Note that interleave is possible only if both the loops operate with the same switching frequency.

The PMBus interleave command for a parallel dual-loop operation would be group identification number = 1, number of devices = 2, interleave order = 0 (for both loop 0 and loop 1), and for interleaved dual operation it would be group identification number = 1, number of devices = 2, interleave order = 0 (loop 0) and interleave order = 1 (loop 1). This section shows the waveforms captured for all the cases mentioned.

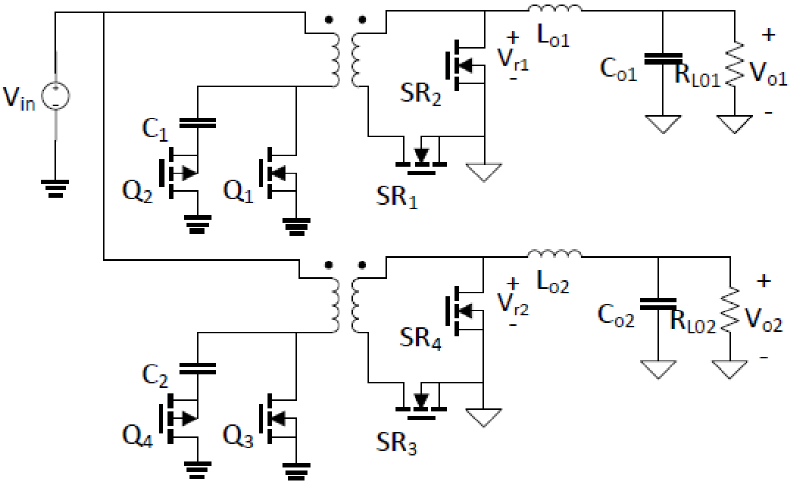


Figure 144 Dual-loop ACF schematic

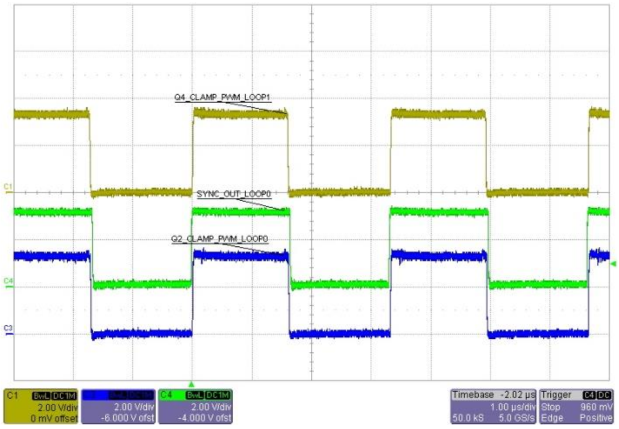
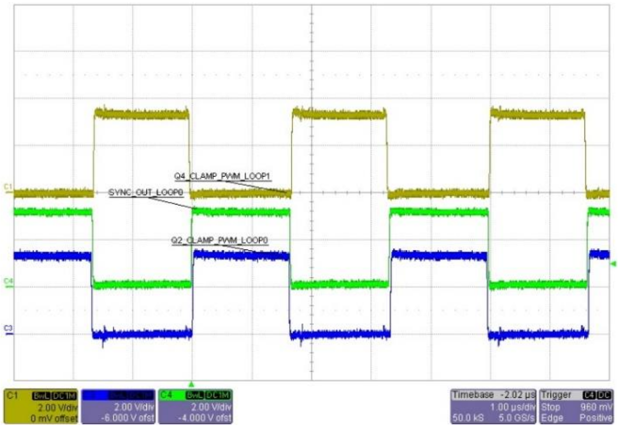
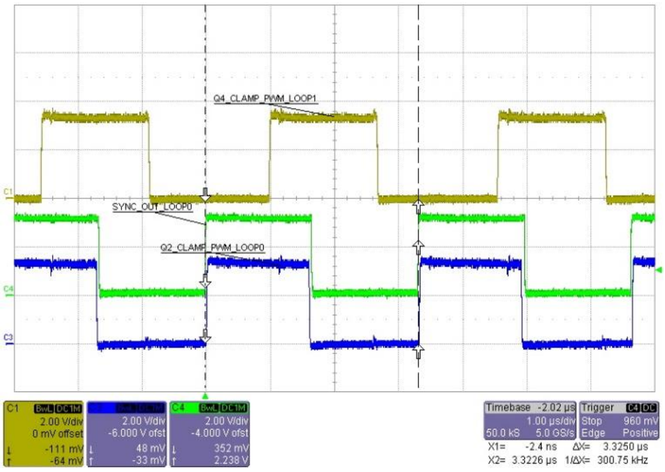
 48 V, 15 A, same f_{sw}, non-interleave	 48 V, 15 A, same f_{sw}, interleave
 48 V, 15 A, different f_{sw}	Channel 1 (yellow) – Q4_Clamp_PWM (loop1), 2 V/div, 1 µs/div – represents PWM control signal of Q4 MOSFET per **Figure 144**. Channel 2 (blue) – Q2_Clamp_FET (loop0), 2 V/div, 1 µs/div – represents PWM control signal of Q2 MOSFET per **Figure 144**. Channel 4 (green) – Sync_Out_Loop0, 2 V/div, 1 µs/div – represents internal sync clock.

Figure 145 Dual-loop waveforms

As shown above, in the case of a different switching frequency in loop 0 and loop 1, the sync-out will be in sync with the loop 0, phase 0 PWM, but asynchronous with the loop 1 PWM.

10.4 Sync-out using different types of modulation

This section demonstrates the sync-out waveforms using an internal sync clock under different PWM modulation techniques.

10.4.1 Trailing edge

Figure 146 is the sync-out waveform using trailing-edge modulation. As in Table 73, the falling edge of the SR2 MOSFET or the rising edge of the primary Q1 MOSFET aligns with the sync-out rising edge. The waveforms are captured using a HBCT topology shown in **Figure 134**. The delay observed in the Q1 rising edge is because of sync block propagation delay and rising edge dead-time of Q1.

Channel 2 (red) – Q1 rise, 2 V/div, 1 µs/div – represents PWM control signal of Q1 MOSFET per **Figure 134**.

Channel 3 (blue) – SR2 fall, 2 V/div, 1 µs/div – represents PWM control signal of SR2 MOSFET per **Figure 134**.

Channel 4 (green) – external sync signal, 2 V/div, 1 µs/div – represents function generator signal.

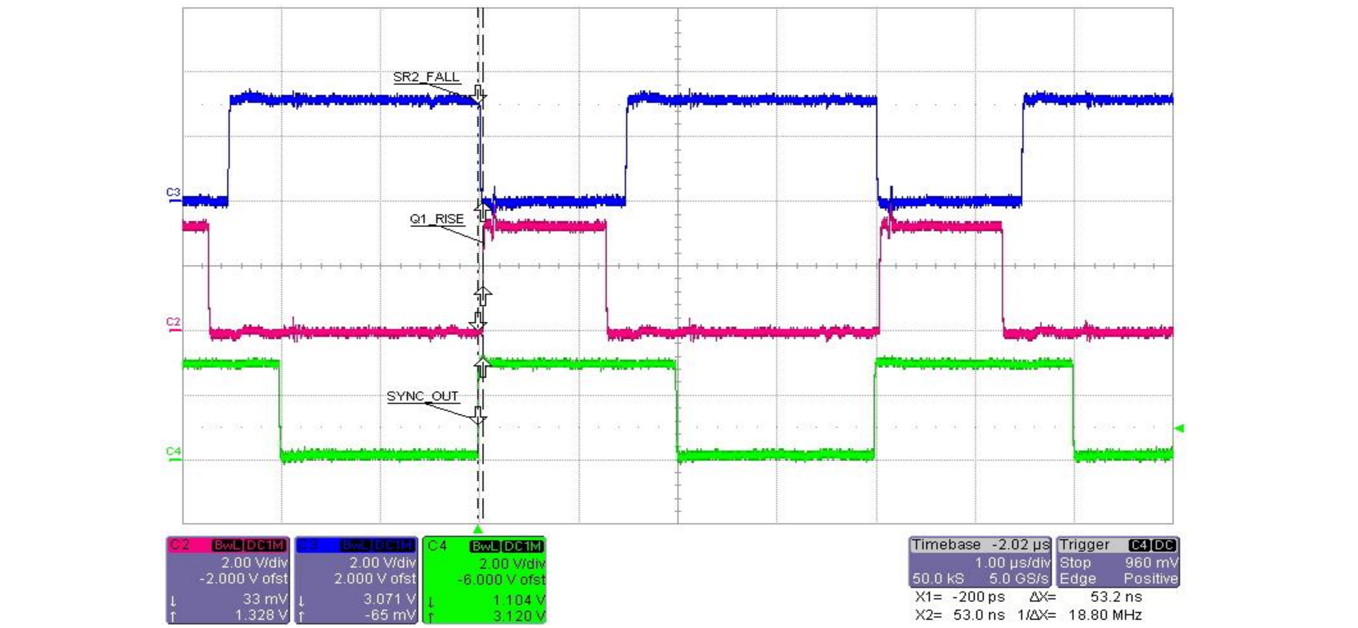


Figure 146 Sync-out trailing edge waveform

10.4.2 Leading edge

Figure 147 is the sync-out waveform using leading-edge modulation. As in Table 73, the rising edge of the SR1 MOSFET or the falling edge of the primary Q2 MOSFET aligns with the sync-out rising edge. The waveforms are captured using a HBCT topology shown in **Figure 134**. The delay observed in the SR1 rising edge is because of sync block propagation delay and rising edge dead-time of SR1.

Channel 2 (red) – Q2 fall, 2 V/div, 1 µs/div – represents PWM control signal of Q2 MOSFET per **Figure 134**.

Channel 3 (blue) – SR1 rise, 2 V/div, 1 µs/div – represents PWM control signal of SR1 MOSFET per **Figure 134**.

Channel 4 (green) – external sync signal, 2 V/div, 1 µs/div – represents function generator signal.

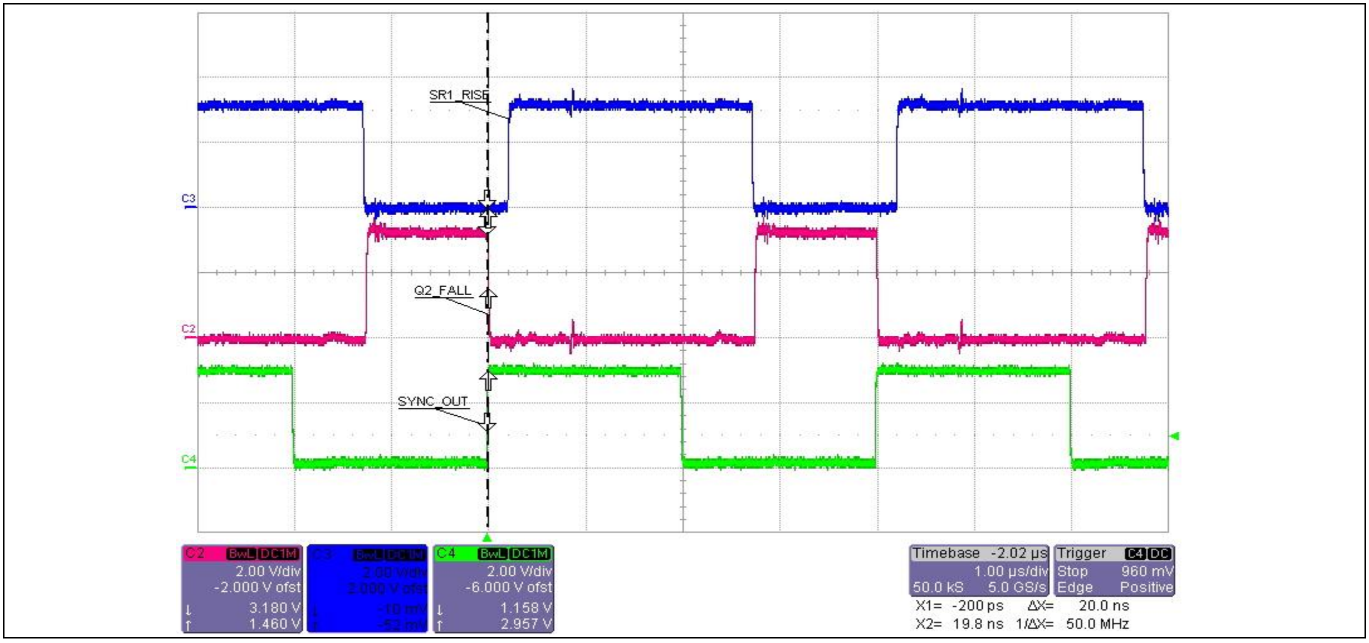


Figure 147 Sync-out for leading edge waveform

10.4.3 Dual edge

Figure 148 demonstrates sync-out waveforms using dual-edge modulation. As in Table 73, the sync-out rising edge is in between the Q2 falling edge and Q1 rising edge. The waveforms are captured using a HBCT topology shown in **Figure 134**. A slight misalignment on the Q1 rising edge is because of sync block propagation delay and rising edge dead-time of Q1.

Channel 2 (red) – Q2 fall, 2 V/div, 1 µs/div – represents PWM control signal of Q2 MOSFET per **Figure 134**.

Channel 3 (blue) – Q1 rise, 2 V/div, 1 µs/div – represents PWM control signal of Q1 MOSFET per **Figure 134**.

Channel 4 (green) – external sync signal, 2 V/div, 1 µs/div – represents function generator signal.

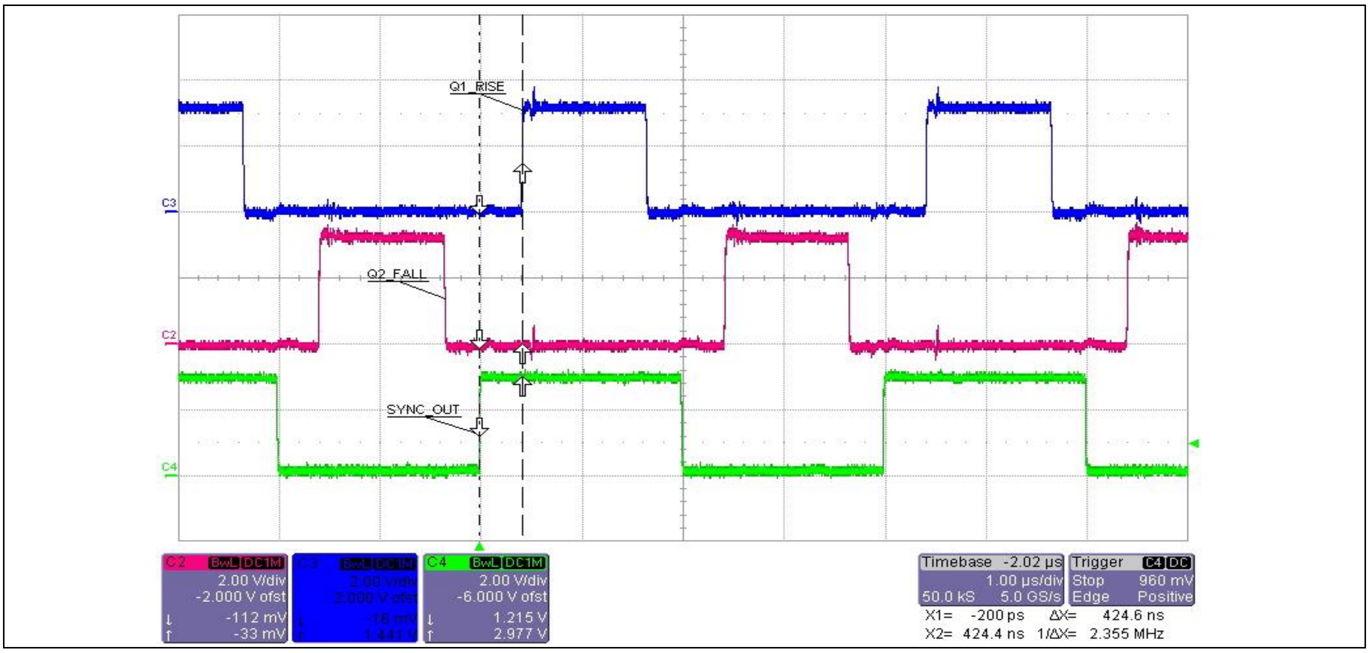


Figure 148 Sync-out dual-edge waveform

10.5 Sync-in range

The sync input signal frequency is limited to +/-6.25 percent of the programmed switching frequency for a successful lock with the sync clock. Once locked, the sync input signal can maintain lock up to +/- 12.5 percent of the programmed frequency.

Please note, the frequency resolution is 20 ns. Setting FREQUENCY_SWITCH to a number doesn't mean the controller will switch at this value. The actual switching frequency is calculated using the following equation:

$$F_{sw} = \frac{1}{(INT\left(\frac{TSW}{20ns}\right)x20ns)}$$

For example,

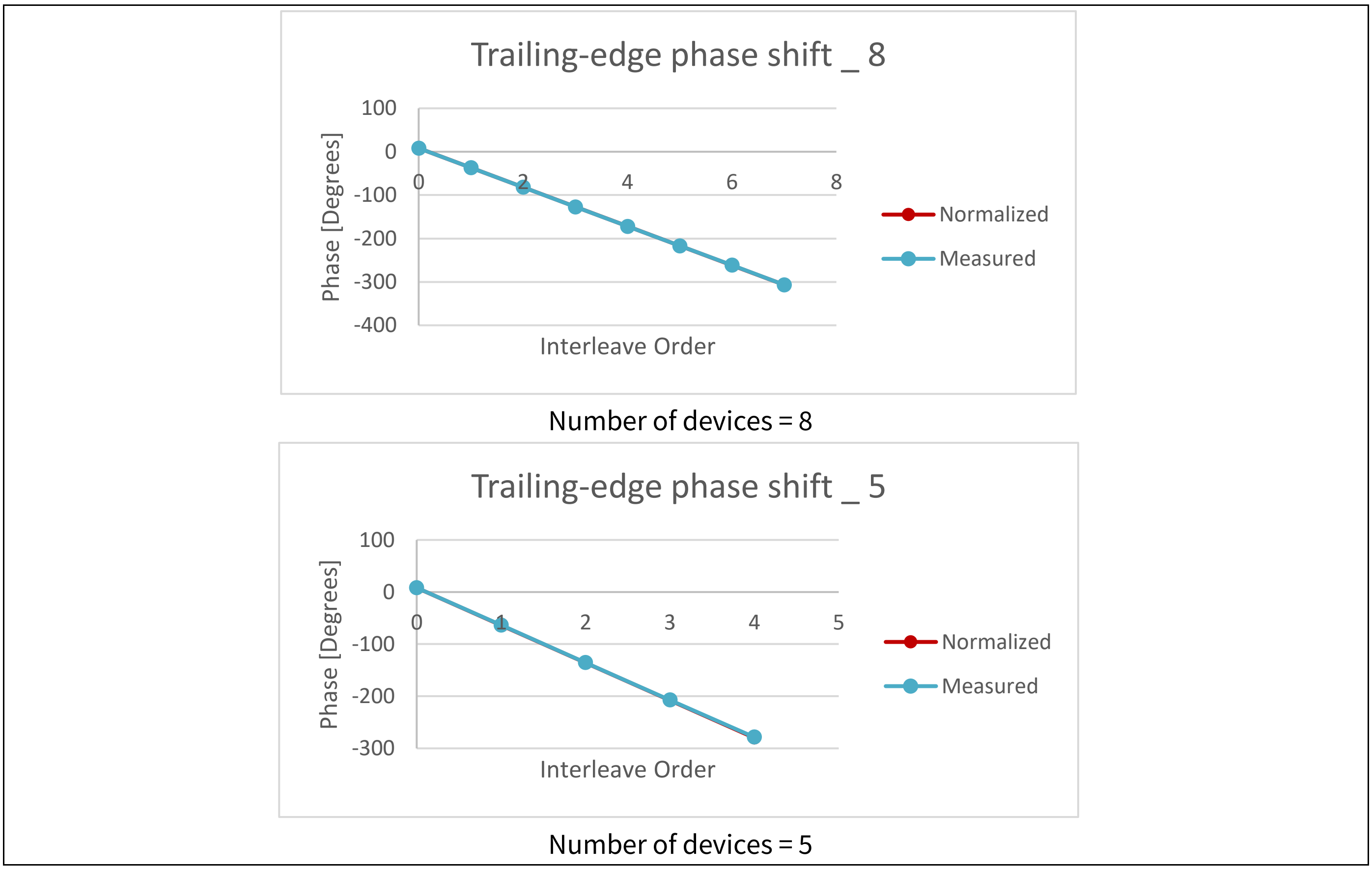
FREQUENCY_SWITCH = 300 kHz, T_{SW} = 3333 ns, $F_{sw} = \frac{1}{(INT\left(\frac{3333ns}{20ns}\right)x20ns)}$ = 301 kHz.

10.6 Phase-shift accuracy

The phase-shift accuracy of the XDPP1100 is 1.40625 degrees. The phase delay of trailing edge modulation was tested with an even and odd set of INTERLEAVE configurations. The number of devices in the group was set to eight and five to verify the phase delay accuracy. The error between normalized and measured data is less than 1 percent.

In **Figure 149**, normalized phase shift = calculated phase shift + error observed from Q1 PWM rising to sync-in rising for interleaved order zero.

Measured phase shift = actual phase shift measurement using oscilloscope.



Number of devices = 8

Number of devices = 5

Figure 149 Phase delay with device number

10.7 Sync-in performance in different modes of converter operation

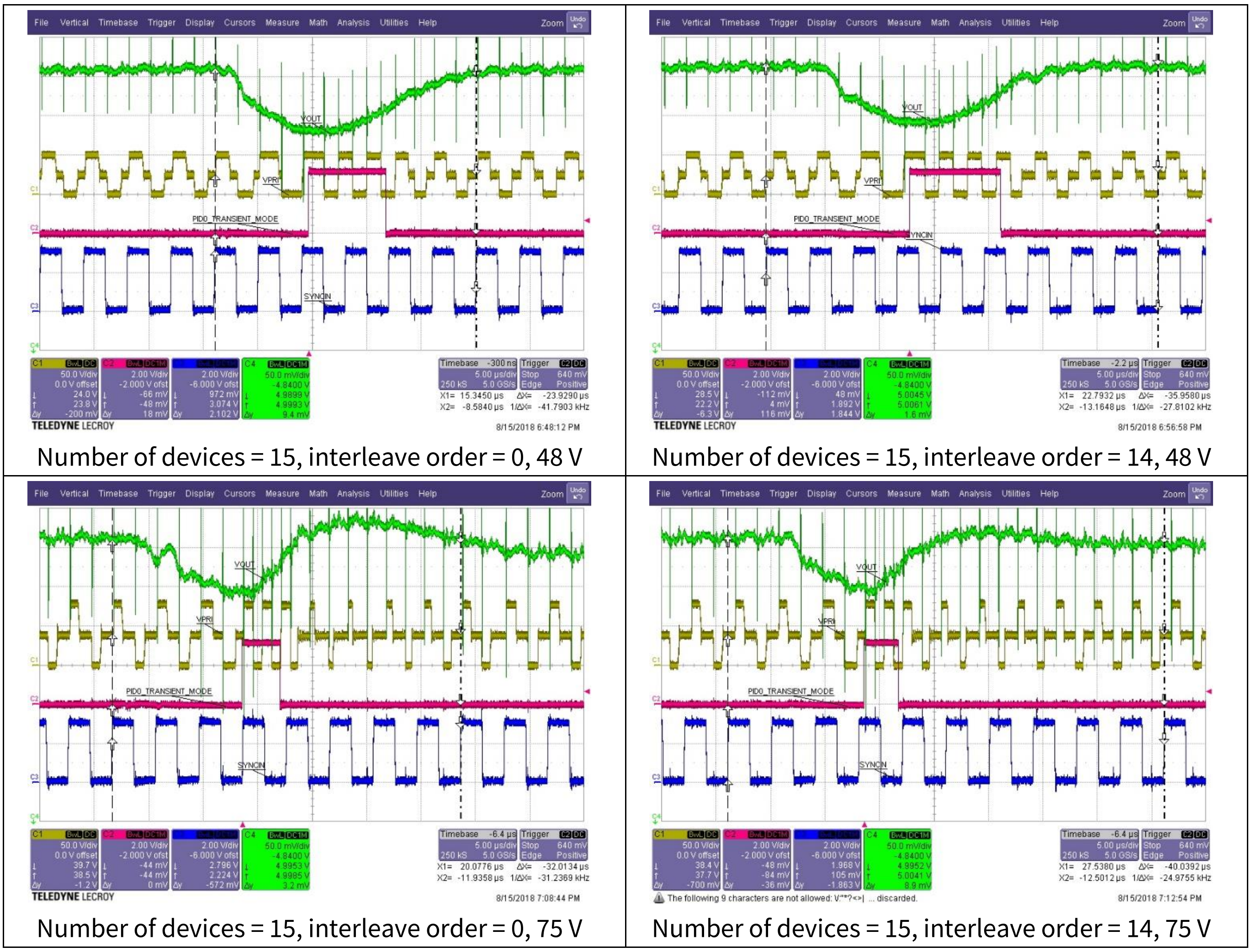
The XDPP1100 usually operates under a fixed-frequency except in FTR or burst mode. This chapter demonstrates the performance of XDP1100 when it is in sync with an external clock while the FTR or burst occurs.

10.7.1 Fast transient

The FTR registers were set as in Table 74. When the system enters FTR mode, it is no longer aligned with the sync clock. When it exits, it starts with 8/7 the nominal switching frequency in order to move the phase of the PWM and slowly reduces to nominal switching frequency until they are aligned. The maximum number of cycles it takes to sync with the external clock after exiting FTR is 7.

Table 74 Register settings to enable FTR capability

Register name	Register value
pid0_verr_entry_thrs	20
pid0_verr_slope_entry_thrs	8
pid0_verr_exit_thrs	3
pid0_verr_slope_exit_thrs	3



Number of devices = 15, interleave order = 0, 48 V

Number of devices = 15, interleave order = 14, 48 V

Number of devices = 15, interleave order = 0, 75 V

Number of devices = 15, interleave order = 14, 75 V

Figure 150 Sync-in test with a step-load transient from 0 to 16.66 A using a 0.3 Ω resistor

Channel 1 (yellow) – V_{pri}, 50 V/div, 20 µs/div – represents voltage across transformer primary.

Channel 2 (red) – I_{OUT}, 2 A /div, 20 µs/div – represents load current.

Channel 3 (blue) – sync-in, 2 V/div, 20 µs/div – represents external sync signal from function generator.

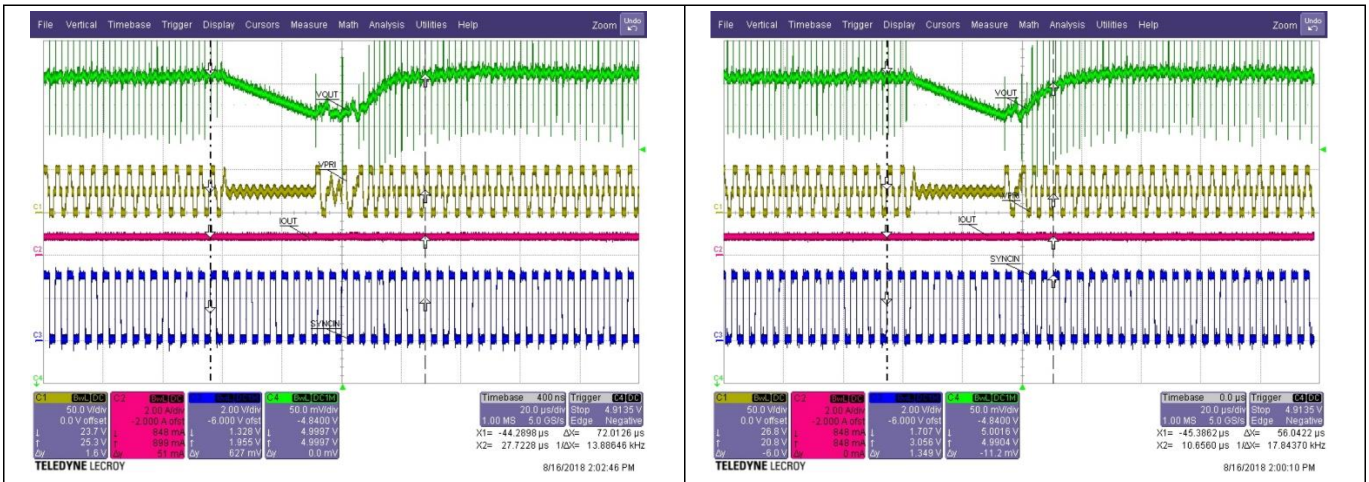
Channel 4 (green) – V_{OUT}, 50 mV/div, 20 µs/div – represents output voltage.

10.7.2 Burst mode

Burst registers were set as in Table 75. When the system enters burst mode, it is no longer aligned with the sync clock. When it exits, it starts with 8/7 the nominal switching frequency in order to move the phase of the PWM and slowly reduces to nominal switching frequency until they are aligned. The maximum number of cycles it takes to sync with the external clock after burst exit is 7.

Table 75 Register values to enable burst mode

Register name	Register value
Power mode	3
pid0_burst_mode_ith	30
pid0_burst_mode_err_thr	10
pid0_burst_reps	1


File Vertical Timebase Trigger Display Cursors Measure Math Analysis Utilities Help Zoom Undo
VOUT
VPRI
IOUT
SYNCIN
C1
C2
C3
C4
C1 BwL DC 50.0 V/div 0.0 V offset ↓ 23.7 V ↑ 25.3 V Δy 1.6 V
C2 BwL DC 2.00 A/div -2.000 A ofst ↓ 848 mA ↑ 899 mA Δy 51 mA
2.00 V/div -6.000 V ofst ↓ 1.328 V ↑ 1.955 V Δy 627 mV
C4 BwL DC1M 50.0 mV/div -4.8400 V ↓ 4.9997 V ↑ 4.9997 V Δy 0.0 mV
Timebase 400 ns 20.0 µs/div 1.00 MS 5.0 GS/s
Trigger C4 DC Stop 4.9135 V Edge Negative
X1= -44.2898 µs ΔX= 72.0126 µs
X2= 27.7228 µs 1/ΔX= 13.88646 kHz
TELEDYNE LECROY
8/16/2018 2:02:46 PM
File Vertical Timebase Trigger Display Cursors Measure Math Analysis Utilities Help Zoom Undo
VOUT
VPRI
IOUT
SYNCIN
C1 BwL DC 50.0 V/div 0.0 V offset ↓ 26.8 V ↑ 20.8 V Δy -6.0 V
C2 BwL DC 2.00 A/div -2.000 A ofst ↓ 848 mA ↑ 848 mA Δy 0 mA
2.00 V/div -6.000 V ofst ↓ 1.707 V ↑ 3.056 V Δy 1.349 V
C4 BwL DC1M 50.0 mV/div -4.8400 V ↓ 5.0016 V ↑ 4.9904 V Δy -11.2 mV
Timebase 0.0 µs 20.0 µs/div 1.00 MS 5.0 GS/s
Trigger C4 DC Stop 4.9135 V Edge Negative
X1= -45.3862 µs ΔX= 56.0422 µs
X2= 10.6560 µs 1/ΔX= 17.84370 kHz
TELEDYNE LECROY
8/16/2018 2:00:10 PM

Number of devices = 15, interleave order = 0, 48 V

Number of devices = 15, interleave order = 14, 48 V

Figure 151 Sync-in test during burst mode operation

Channel 1 (yellow) – V_{pri}, 50 V/div, 20 µs/div – represents voltage across transformer primary.

Channel 2 (red) – I_{OUT}, 2 A/div, 20 µs/div – represents load current.

Channel 3 (blue) – sync-in, 2 V/div, 20 µs/div – represents external sync signal from function generator.

Channel 4 (green) – V_{OUT}, 50 mV/div, 20 µs/div – represents output voltage.

11 Temperature sense

The XDPP1100 supports up to two external temperature sensors and one internal temperature sensor. The external temperature can be sensed by connecting a 47 kΩ NTC resistor in parallel with a 12 kΩ resistor to the TSEN or BTSEN pin. A 100 µA current source is enabled when the corresponding current DAC is set to 1 (see the temperature sense register description **ts_tsidac_antc_sel** and **ts_tsidac_bntc_sel**). The voltage at the TSEN pin is determined by the 100 µA current and the total equivalent resistance at the pin. This voltage is sensed by TS ADC (see chapter 1.5) and decoded by FW through a lookup table.

The temperature lookup table that is coded in the XDPP1100 is optimized for the recommended NTC resistor Murata NCP15WB473F03RC or Panasonic ERT-J0EP473J. The accuracy of temperature sense is +/- 5°C in temperature range 0°C to 125°C. Below 0°C, the NTC value is very large and the total equivalent resistance is mainly contributed by the 12 kΩ resistor and is flattened. Thus, the accuracy is reduced at low temperatures.

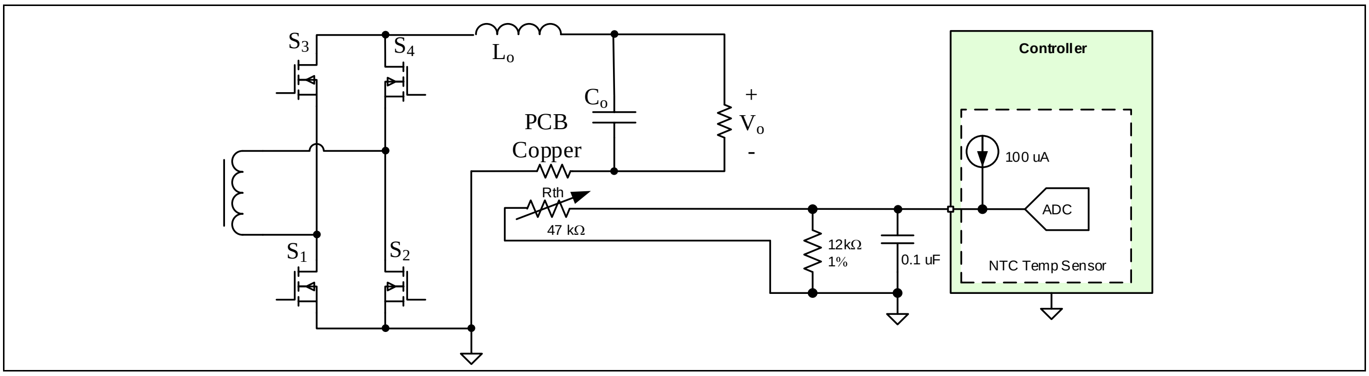


Figure 152 NTC temperature sensing

Use of other temperature sensor types (e.g., PTC resistor or diode) is supported via FW patch with a custom temperature lookup table. The input voltage range of the TS ADC is 0 V to 1.2 V. The temperature sense device should be selected to work in this range.

The internal temperature sensor is a proportional to absolute temperature (PTAT) based sensor that indicates the junction temperature of the controller. The accuracy of the internal temperature sensor is +/- 5°C over the -40°C to 150°C range.

11.1 Temperature sense registers description

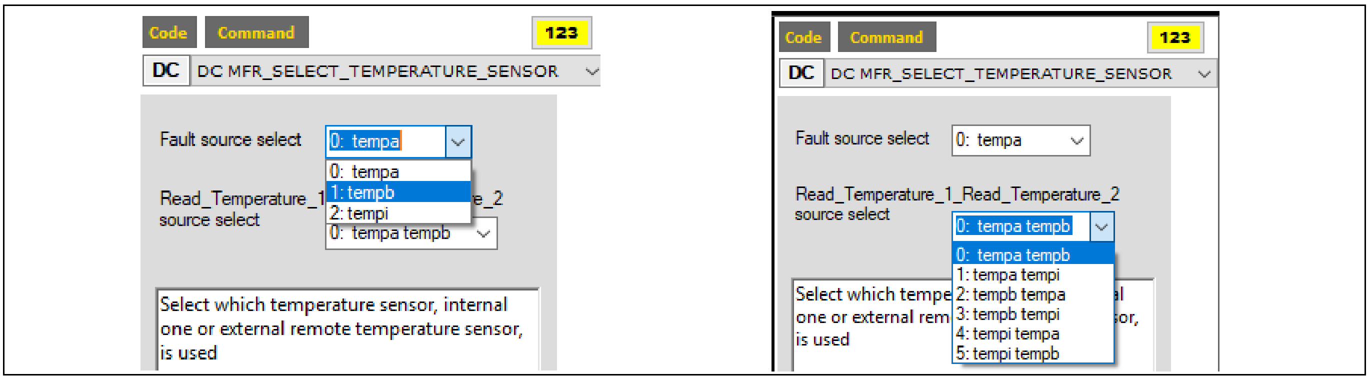
Table 76 Temperature sense registers

Name	Address	Bits	Description
TSEN peripheral			
atsen_meas_en	7000_4C00_H	[2:2]	TS ADC ATSEN measurement enable. When enabled, the TS ADC will measure the ATSEN input when selected by ts_muxmode and tx_muxctrl2. When disabled, no ATSEN measurement will occur, even if selected by ts_muxmode and ts_muxctrl2.
btsen_meas_en	7000_4C00_H	[3:3]	TS ADC BTSEN measurement enable. When enabled, the TS ADC will measure the BTSEN input when selected by ts_muxmode and tx_muxctrl2. When disabled, no BTSEN measurement will occur, even if selected by ts_muxmode and ts_muxctrl2.

Name	Address	Bits	Description
itsen_meas_en	7000_4C00_H	[4:4]	TS ADC internal temperature (ITSEN) measurement enable. When enabled, the TS ADC will measure ITSEN when selected by ts_muxmode and tx_muxctrl2. When disabled, no ITSEN measurement will occur, even if selected by ts_muxmode, ts_muxctrl1 and ts_muxctrl2.
ts_tsidac_antc_sel	7000_4C00_H	[6:6]	ATSEN output current source enable. This current source should be enabled when using the ATSEN input to measure an NTC or PTC temperature sense element. The current source may be disabled to use the ATSEN input as a general-purpose ADC input.
ts_tsidac_bntc_sel	7000_4C00_H	[7:7]	ATSEN output current source enable. This current source should be enabled when using the ATSEN input to measure an NTC or PTC temperature sense element. The current source may be disabled to use the ATSEN input as a general-purpose ADC input.
ptat_pwl_slope	7000_4C04_H	[27:16]	Internal temperature sense (ITSEN) piecewise linear slope term. Should be set to 990.
ptat_0c_code	7000_4C08_H	[10:0]	Internal temperature sense (ITSEN) 0C code. Should be set to 926.

11.2 Temperature sense PMBus command

The XDPP1100 GUI allows the user to configure which temperature sensor is used for temperature fault protection. The MFR_SELECT_TEMPERATURE_SENSOR PMBus command shown in **Figure 153** demonstrates how to select the temperature sensor for the fault source and for the telemetry READ_TEMPERATURE source.



DC MFR_SELECT_TEMPERATURE_SENSOR

Fault source select: 0: tempa

Read_Temperature_1 Read_Temperature_2 source select: 0: tempa tempb

Select which temperature sensor, internal one or external remote temperature sensor, is used

Figure 153 MFR_SELECT_TEMPERATURE_ SENSOR command

The temperature assigned to READ_TEMPERATURE_1 is used for output current temperature compensation when the I_{OUT} temperature compensation is enabled by FW_CONFIG_REGULATION bit [9].

12 Fault protections

With high-performance ADCs, the XDPP1100 implements all the fault protections without using external comparators. This chapter describes how the XDPP1100 implements the fault protections on input/output voltage, input/output current, input/output power and internal /external temperature fault. Besides the standard fault protections mentioned above, the XDPP1100 also offers cycle-by-cycle PCL, SCP, voltage open sense fault detection, current ADC tracking fault detection, and flux balance fault protection.

12.1 Introduction to XDPP1100 fault management

The XDPP1100 fault detection is implemented in HW and the fault response is managed by FW. The combination of HW and FW provides fast protection with flexible fault response.

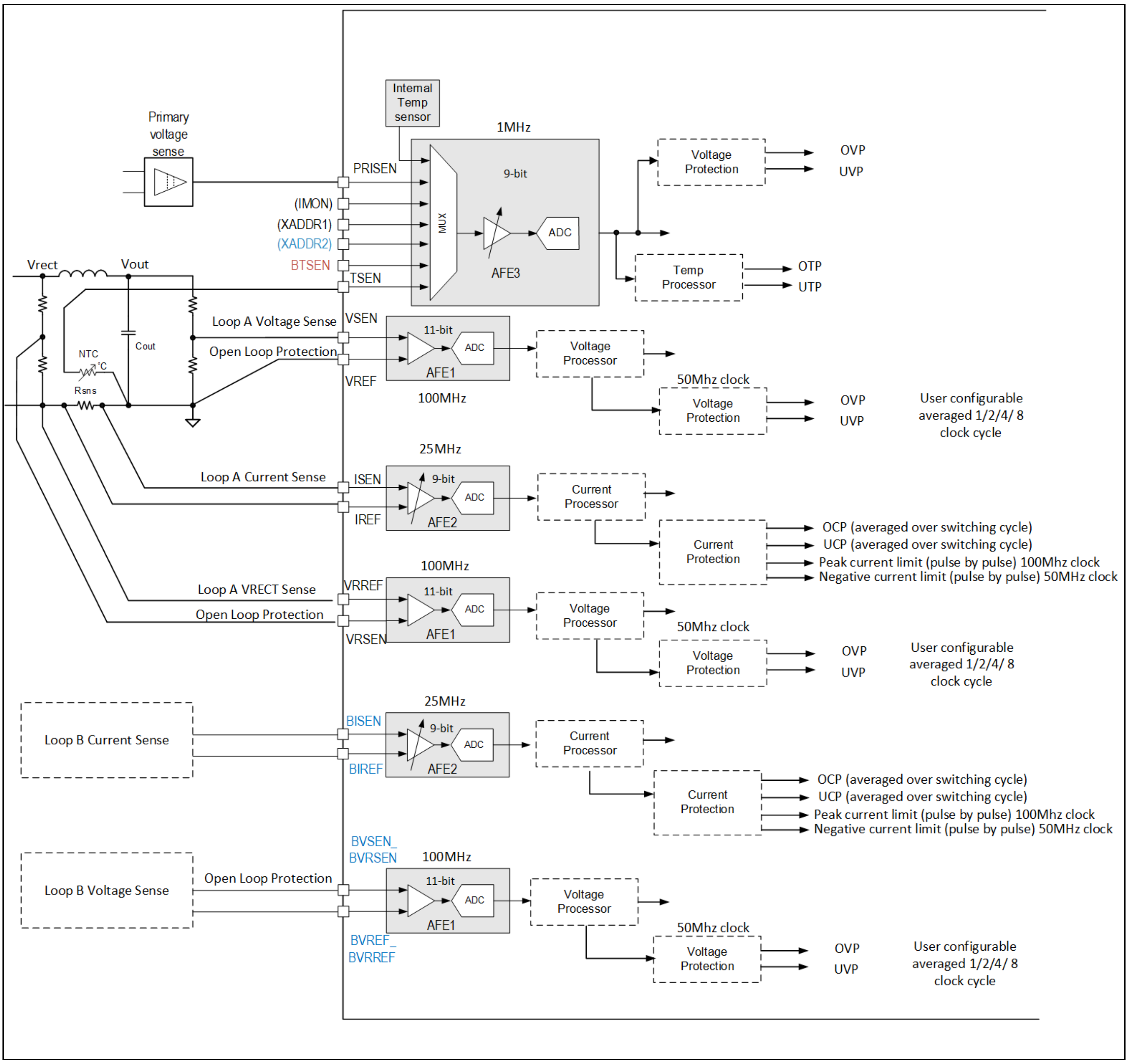


Figure 154 XDPP1100 fault protection block diagram for a typical application

Figure 154 shows the XDPP1100 block diagram of fault detections of a typical application. The voltage sense ADC (VSEN, VRSEN and BVSEN_BVRSEN) is a 100 MHz, 11-bit ADC. The ADC is clocked at 100 MHz and a voltage sense front-end amplifier processes the voltage at 50 MHz. The current sense ADC (ISEN, BISEN) is 25 MHz, 9-bit ADC and is interpolated to 13-bit through current emulation. In most applications, the VSEN ADC is used for output voltage sensing, the VRSEN ADC is used for input voltage sensing (via V_{RECT} sensing), and the ISEN ADC is used for output current sensing. For XDPP1100-Q040, which supports dual-loop control, the BVSEN and BISEN can be used for the second loop voltage and current sensing respectively.

The input voltage sense, input current sense and output current sense in **Figure 154** show the typical example configuration supported by the XDPP1100. The XDPP1100 also has the flexibility of mapping either VRSEN or BVRSEN for input voltage sense, and mapping either ISEN or BISEN for output current sense. The source signals are selected by the registers listed in Table 77. Please note that the output voltage sense uses a dedicated VADC per loop, thus it is not selectable. The output voltage of loop 0 must be sensed by VSEN ADC, and the output voltage of loop 1 must be sensed by BVSEN ADC.

The BVSEN_BVRSEN ADC is a dual-function ADC. It can be configured for V_{OUT} sensing or V_{RECT} (V_{IN}) sensing. The mode is selected by register **vsp2_vrs_sel**. The telemetry source selections of I_{IN}, I_{OUT} and V_{IN} are configurable for both loop 0 and loop 1. For example, **tlm0_iin_src_sel** configures the input current source of loop 0, and **tlm1_iout_src_sel** configures the output current source of loop 1.

Table 77 Telemetry I_{IN}, I_{OUT}, V_{IN} source select

Register name	Description
tlm_iin_src_sel	Input current telemetry source select. 0 = measured on ISEN input 1 = measured on BISEN input 2, 3 = estimated input current based on output current
tlm_iout_src_sel	Output current telemetry source select. 0 = measured on ISEN input 1 = measured on BISEN input 2, 3 = sum of ISEN and BISEN inputs (dual-phase of interleaved topologies)
tlm_vin_src_sel	Input voltage telemetry source select. 0 = VRSEN. Secondary V_{RECT} sense, **vrs_init** prior to start-up 1 = BVSEN_BVRSEN. Secondary V_{RECT} sense, **vrs_init** prior to start-up 2 = loop 0 V_{OUT}. Select on loop 1 when loop 1 V_{IN} provided by loop 0 V_{OUT} (e.g., post-buck). 3 = TS ADC V_{IN}. Non-pulsed/primary V_{IN} sense via telemetry ADC (PRISEN) 4 = tlm_vin_force. Forced V_{IN} via FW (e.g., FW override of HW computation) 5 = VRSEN. Secondary V_{RECT} sense, **0 V** prior to start-up. Select on loop 1 when sharing loop 0 V_{RECT} sense 6 = VRSEN. Non-pulsed/primary V_{IN} sense 7 = BVSEN_BVRSEN. Non-pulsed/primary V_{IN} sense
vsp1_vrs_sel	VRSEN input ADC rectification voltage sense (VRS) mode select. 0 = general-purpose ADC mode 1 = V_{RECT} sense (VRS) mode
vsp2_vrs_sel	BVSEN_BVRSEN input ADC rectification voltage sense (VRS) mode select. 0 = V_{OUT} sense (VS) mode 1 = V_{RECT} sense (VRS) mode

For output voltage sensing, i.e. VSEN and BVSEN (**vsp2_vrs_sel** = 0), the ADC has overvoltage and undervoltage detection running at 50 MHz (20 ns). The latency from the fault detection to PWM response is less than 30 ns, giving total fault response time of less than 50 ns. The VADC offers one, two, four or eight clock cycles of averaging, allowing the user to configure the sensitivity of the fault detection. The fault counters for overvoltage and undervoltage are configured independently. For example, setting the output overvoltage fault count to two clock cycles would only assert the OVP fault when two consecutive samples exceed the fault threshold. Please note that the fault response will be delayed when the fault count is set to more than one clock cycle. Each clock cycle will add 20 ns to the overall delay.

For input voltage sensing, the protection is detected every switching cycle. It allows users to configure the number of consecutive switching cycles by which the input voltage must exceed the threshold in order to assert the fault. The number of consecutive cycles can be set to one, two, four, or eight.

Each current ADC offers over-current and under-current detection. OCP and UCP are based on average current, which is averaged over the switching frequency of the power supply converter. OCP and UCP fault assertion requires the current to exceed the fault threshold for a user-configurable consecutive number of samples. The number of consecutive samples can be programmed from one to 32 switching cycles.

The XDPP1100 offers a second-level I_{OUT} fast OCP. The MFR_IOUT_OC_FAST_FAULT defines the fast over-current threshold and response. The fast over-current is sensed before the I_{OUT} telemetry LPF. It provides faster fault response than the standard IOUT_OC_FAULT, which is sensed after the LPF.

The third-level OCP is achieved by SCP, which trips on a single 25 MHz sample by the CS ADC.

The current ADC also provides cycle-by-cycle PCL. The positive PCL operates on a 100 MHz clock, and the negative PCL operates on a 50 MHz clock. The PCL response time is less than 50 ns from the current sense input to PWM output.

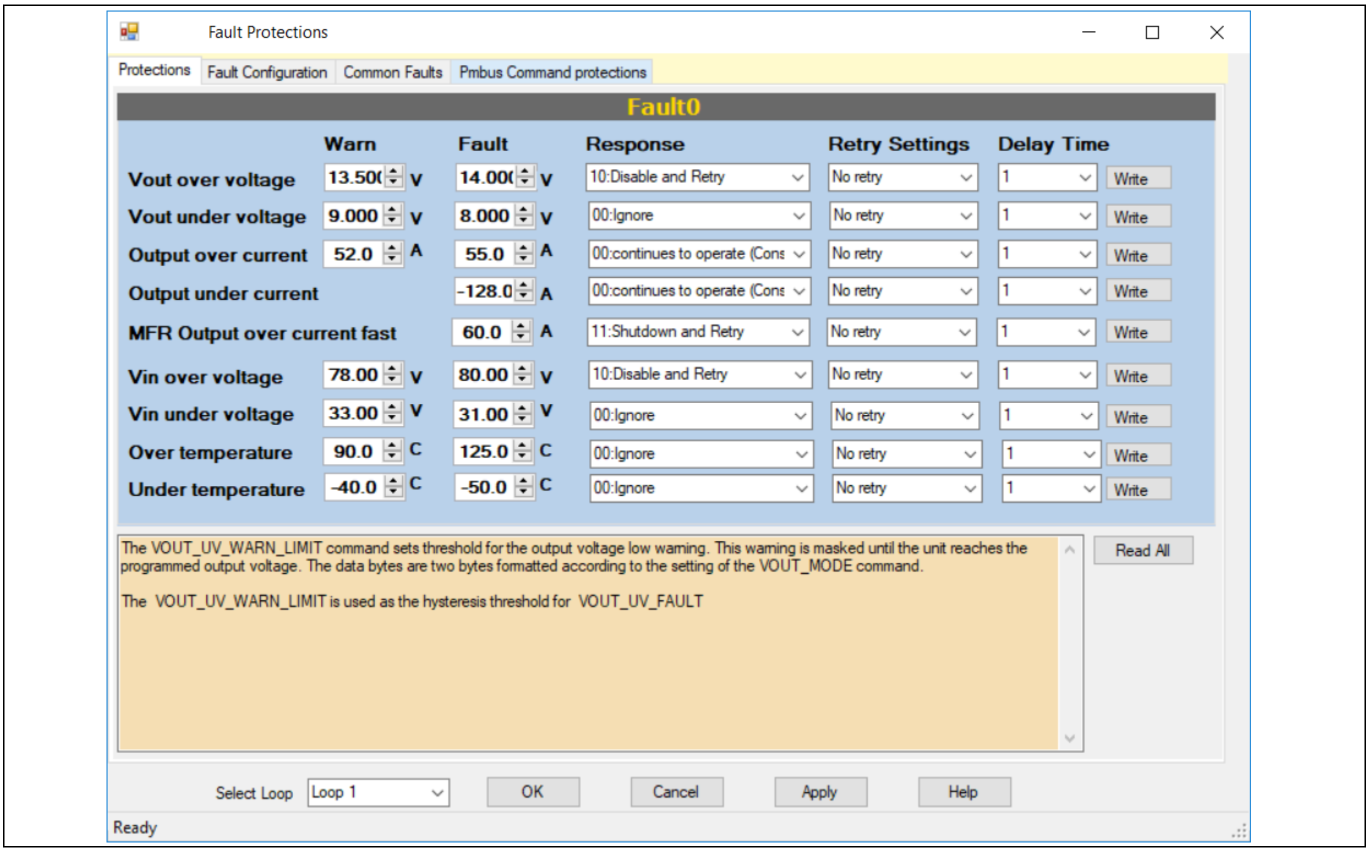
The XDPP1100 telemetry ADC (TS ADC) is a 1 MHz 9-bit ADC. This ADC has eight muxed inputs and can be configured to digitize voltage and temperature (see the list below). The input voltage can be sensed by either the VADC or the TS ADC (PRISEN). Use of the 100 MHz VADC is recommended for speed and performance. The temperature can be sensed using an external thermal resistor or by IC internal temperature sensor. The input voltage range of the TS ADC is 0 V to 1.2 V.

- PRISEN, primary voltage sensing
- IMON, current monitoring and active current sharing
- ATSEN, temperature sense A
- BTSEN, temperature sense B
- XADDR1, address offset 1
- XADDR2, address offset 2
- ITSEN, internal temperature sense

Overtemperature fault protection can be configured to use either external ATSEN, BTSEN or the internal ITSEN. The external temperature sense can either be NTC resistor (default) or PTC resistor. The default ATSEN and BTSEN temperature sense use 47 kΩ NTC in parallel with a 12 kΩ fixed resistor. Please see the XDPP1100 datasheet for the connection of external resistors. A FW patch with user-defined temperature lookup table allows the XDPP1100 to support any kind of thermal sense.

12.2 XDPP1100 GUI design tool

The XDPP1100 GUI design tool “Fault Protections” allows the user to configure the fault-related PMBus commands and registers including the fault thresholds and fault response (**Figure 155**).



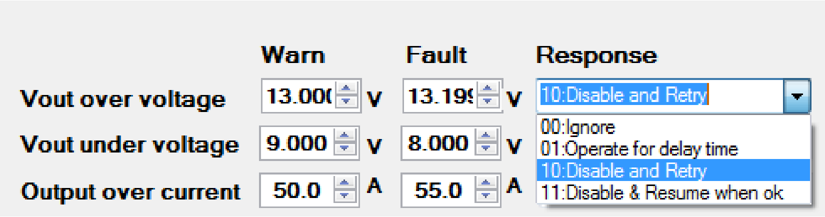
	Warn		Fault		Response	Retry Settings	Delay Time	
Vout over voltage	13.50(	V	14.00(	V	10:Disable and Retry	No retry	1	Write
Vout under voltage	9.000	V	8.000	V	00:Ignore	No retry	1	Write
Output over current	52.0	A	55.0	A	00:continues to operate (Cons	No retry	1	Write
Output under current			-128.0	A	00:continues to operate (Cons	No retry	1	Write
MFR Output over current fast			60.0	A	11:Shutdown and Retry	No retry	1	Write
Vin over voltage	78.00	V	80.00	V	10:Disable and Retry	No retry	1	Write
Vin under voltage	33.00	V	31.00	V	00:Ignore	No retry	1	Write
Over temperature	90.0	C	125.0	C	00:Ignore	No retry	1	Write
Under temperature	-40.0	C	-50.0	C	00:Ignore	No retry	1	Write

The VOUT_UV_WARN_LIMIT command sets threshold for the output voltage low warning. This warning is masked until the unit reaches the programmed output voltage. The data bytes are two bytes formatted according to the setting of the VOUT_MODE command.

The VOUT_UV_WARN_LIMIT is used as the hysteresis threshold for VOUT_UV_FAULT

Figure 155 **The XDPP1100 design tool – Faults & Protections**

For each fault response, choose the desired response (**Figure 156**), retry setting (**Figure 157**) and delay time (**Figure 158**) from the drop-down list.



	Warn		Fault		Response
Vout over voltage	13.00(	V	13.19(	V	10:Disable and Retry
Vout under voltage	9.000	V	8.000	V	
Output over current	50.0	A	55.0	A	

Figure 156 **Select fault response**

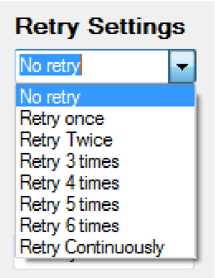
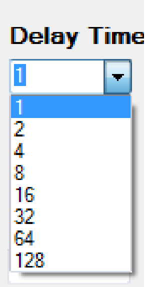

Retry Settings: No retry

Figure 157 **Select fault retry number**



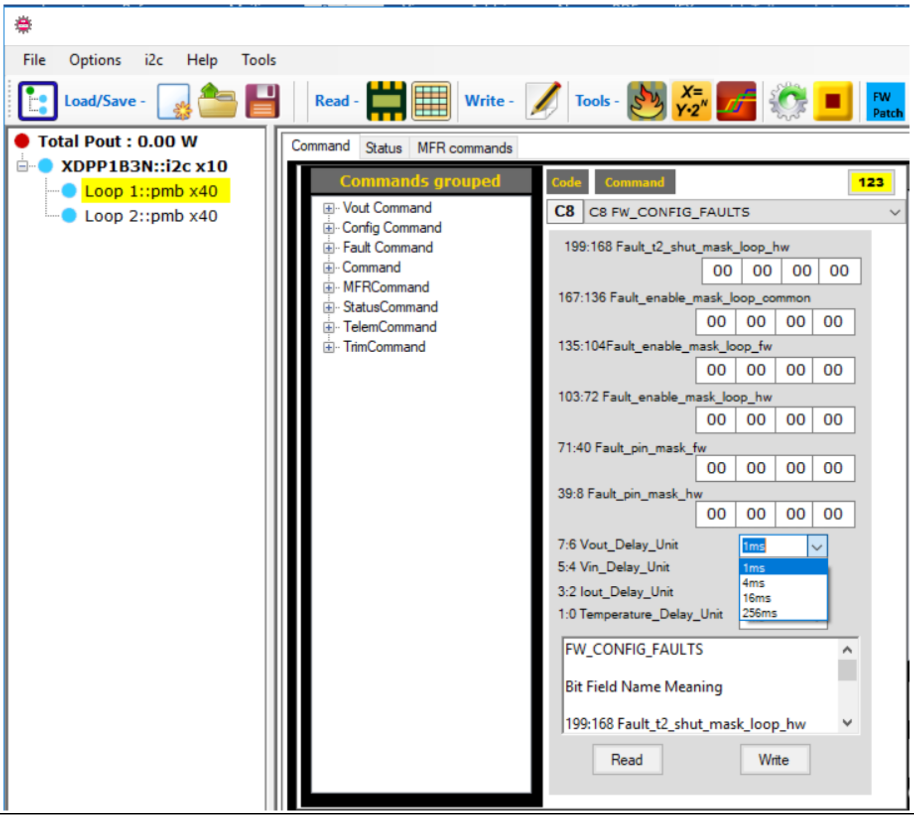
Delay Time

1

1
2
4
8
16
32
64
128

Figure 158 Select fault delay time

Fault delay time together with fault delay unit define the fault operation delay and retry delay. The delay is equal to Fault_Delay_Time * Fault_Delay_Unit. Fault delay unit is programmed by PMBus command 0xC8 FW_CONFIG_FAULTS. The delay unit of V_{OUT}, V_{IN}, I_{OUT} and the temperature are independently configurable with a choice of 1 ms, 4 ms, 16 ms and 256 ms (**Figure 159**).



Total Pout : 0.00 W
XDPP1B3N::i2c x10
Loop 1::pmb x40
Loop 2::pmb x40

C8 C8 FW_CONFIG_FAULTS

199:168 Fault_t2_shut_mask_loop_hw 00 00 00 00
167:136 Fault_enable_mask_loop_common 00 00 00 00
135:104Fault_enable_mask_loop_fw 00 00 00 00
103:72 Fault_enable_mask_loop_hw 00 00 00 00
71:40 Fault_pin_mask_fw 00 00 00 00
39:8 Fault_pin_mask_hw 00 00 00 00
7:6 Vout_Delay_Unit 1ms
5:4 Vin_Delay_Unit
3:2 Iout_Delay_Unit
1:0 Temperature_Delay_Unit

1ms
4ms
16ms
256ms

FW_CONFIG_FAULTS

Bit Field Name Meaning

199:168 Fault_t2_shut_mask_loop_hw

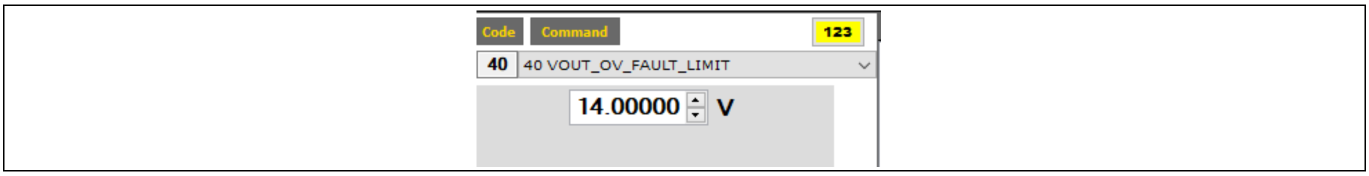
Figure 159 Program fault delay unit

For example, output over-current response is programmed to be “continue to operate for delay time”. With delay time number = 64, Iout_Delay_Unit = 4 ms, the delay time is 64 x 4 ms = 256 ms. Under an output over-current situation, the converter is going to clamp the output current at the value set by IOUT_OC_FAULT_LIMIT for 256 ms and then shut down.

The maximum delay time supported by the XDPP1100 is 128 x 256 ms = 32.768 s.

The input over-current delay unit (not listed in FW_CONFIG_FAULTS) follows Vin_Delay_Unit.

Each individual fault threshold and response can also be configured by GUI in the PMBus section (**Figure 160**).

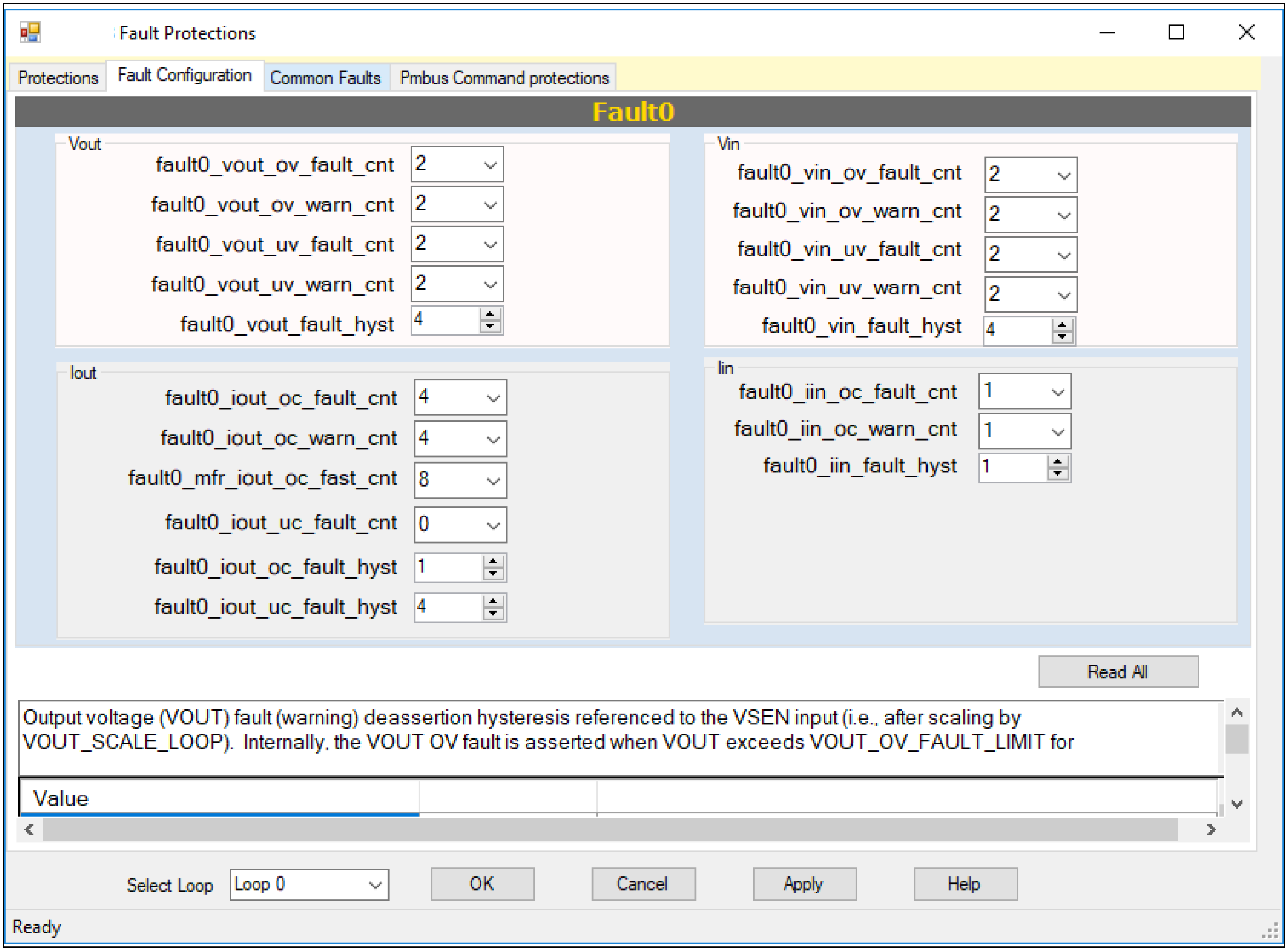


40 | 40 VOUT_OV_FAULT_LIMIT

14.00000 V

Figure 160 Individual PMBus command to configure fault

The fault count number and hysteresis is configured in the XDPP1100 GUI design tool “Fault Protections”, “Fault Configuration” tab (**Figure 161**). Hover the mouse over the register to see the description of each register. Details of each register will be explained in individual fault chapters.



Vout

fault0_vout_ov_fault_cnt	2
fault0_vout_ov_warn_cnt	2
fault0_vout_uv_fault_cnt	2
fault0_vout_uv_warn_cnt	2
fault0_vout_fault_hyst	4

Vin

fault0_vin_ov_fault_cnt	2
fault0_vin_ov_warn_cnt	2
fault0_vin_uv_fault_cnt	2
fault0_vin_uv_warn_cnt	2
fault0_vin_fault_hyst	4

Iout

fault0_iout_oc_fault_cnt	4
fault0_iout_oc_warn_cnt	4
fault0_mfr_iout_oc_fast_cnt	8
fault0_iout_uc_fault_cnt	0
fault0_iout_oc_fault_hyst	1
fault0_iout_uc_fault_hyst	4

Iin

fault0_iin_oc_fault_cnt	1
fault0_iin_oc_warn_cnt	1
fault0_iin_fault_hyst	1

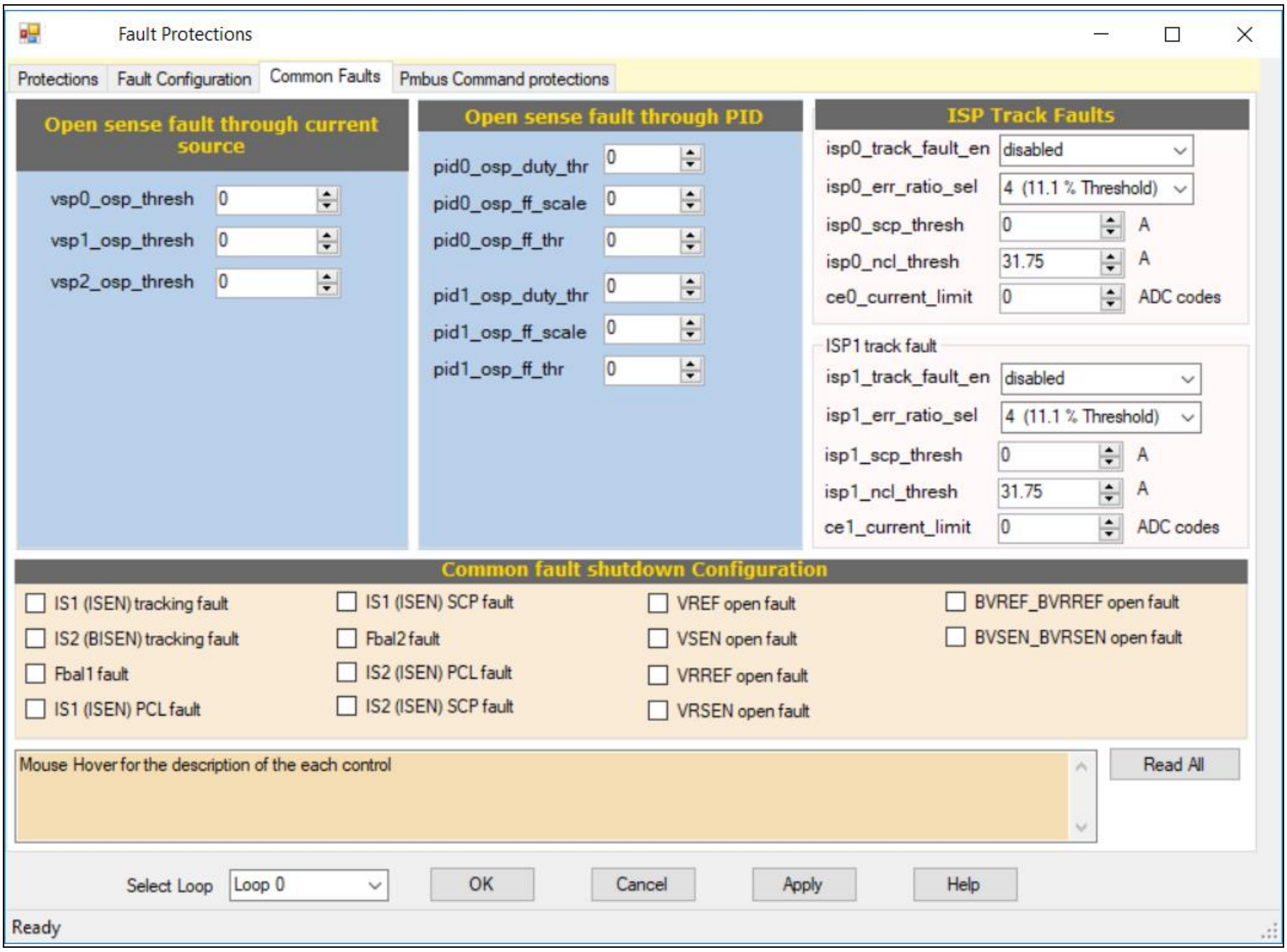
Output voltage (VOUT) fault (warning) deassertion hysteresis referenced to the VSEN input (i.e., after scaling by VOUT_SCALE_LOOP). Internally, the VOUT OV fault is asserted when VOUT exceeds VOUT_OV_FAULT_LIMIT for

Value

Select Loop: Loop 0

Figure 161 Fault registers configuration tool

The XDPP1100 has many unique fault protections, such as loop open sense fault, current sense tracking fault, SCP, positive and negative cycle-by-cycle current limit, and flux balancing fault. These fault protections are grouped as common faults. Common fault configuration can be done in the “Common Faults” tab. Details of common fault protections are explained in chapter 12.9.2.



Fault Protections

Protections | Fault Configuration | Common Faults | Pmbus Command protections

Open sense fault through current source

vsp0_osp_thresh: 0
vsp1_osp_thresh: 0
vsp2_osp_thresh: 0

Open sense fault through PID

pid0_osp_duty_thr: 0
pid0_osp_ff_scale: 0
pid0_osp_ff_thr: 0
pid1_osp_duty_thr: 0
pid1_osp_ff_scale: 0
pid1_osp_ff_thr: 0

ISP Track Faults

isp0_track_fault_en: disabled
isp0_err_ratio_sel: 4 (11.1 % Threshold)
isp0_scp_thresh: 0 A
isp0_ncl_thresh: 31.75 A
ce0_current_limit: 0 ADC codes

ISP1 track fault

isp1_track_fault_en: disabled
isp1_err_ratio_sel: 4 (11.1 % Threshold)
isp1_scp_thresh: 0 A
isp1_ncl_thresh: 31.75 A
ce1_current_limit: 0 ADC codes

Common fault shutdown Configuration

- ☐ IS1 (ISEN) tracking fault
- ☐ IS2 (BISEN) tracking fault
- ☐ Fbal1 fault
- ☐ IS1 (ISEN) PCL fault
- ☐ IS1 (ISEN) SCP fault
- ☐ Fbal2 fault
- ☐ IS2 (ISEN) PCL fault
- ☐ IS2 (ISEN) SCP fault
- ☐ VREF open fault
- ☐ VSEN open fault
- ☐ VRREF open fault
- ☐ VRSEN open fault
- ☐ BVREF_BVRREF open fault
- ☐ BVSEN_BVRSEN open fault

Mouse Hover for the description of the each control

Figure 162 Common fault configuration tool

12.3 Output OVP/UVP

12.3.1 V_{OUT} OVP/UVP PMBus commands

Table 78 describes the PMBus commands relevant to the V_{OUT} OVP and UVP. The unit of the voltage limit thresholds is volts. The data format of the threshold commands is set by VOUT_MODE. The maximum output voltage (VOUT_COMMAND) can be configured to 81.92 V if the user would like to use the feed-forward feature. It is possible to use the XDPP1100 for higher output voltage; a FW patch is required to get around the FF limitation.

Table 78 V_{OUT} OVP/UVP relevant PMBus command descriptions

Command name	Command code	Description
VOUT_OV_FAULT_LIMIT	40_H	This command sets the output overvoltage fault threshold.
VOUT_OV_FAULT_RESPONSE	41_H	This command instructs the device on what action to take in response to an output overvoltage fault. See section 12.3.3 for V_{OUT} OVP fault response.
VOUT_OV_WARN_LIMIT	42_H	This command sets the output overvoltage warning threshold. This value is typically less than the output overvoltage fault

Command name	Command code	Description
		threshold. It is used as the hysteresis threshold for VOUT_OV_FAULT.
VOUT_UV_WARN_LIMIT	43_H	This command sets the output undervoltage warning threshold. This value is typically higher than the output undervoltage fault threshold. This warning is masked until the unit reaches the programmed output voltage. It is used as the hysteresis threshold for VOUT_UV_FAULT.
VOUT_UV_FAULT_LIMIT	44_H	This command sets the output undervoltage fault threshold. The VOUT_UV_FAULT is masked during start-up until the unit reaches the programmed output voltage.
VOUT_UV_FAULT_RESPONSE	45_H	This command instructs the device on what action to take in response to an output undervoltage fault. See section **12.3.3** for V_{OUT} UVP fault response.

12.3.2 V_{OUT} OVP/UVP registers

Table 79 describes the registers that are relevant to the V_{OUT} OVP and UVP.

Table 79 V_{OUT} OV/UV relevant register descriptions

Name	Address (loop 0/1)	Bits	Description
Fault peripheral			
fault_vout_ov_fault_cnt	7000_3C00_H 7000_4000_H	[1:0]	Output overvoltage fault count. Defines the number of consecutive 50 MHz samples by which the output voltage must exceed the fault threshold in order to assert a fault. 0 = 1 sample, 1 = 2 samples 2 = 4 samples, 3 = 8 samples
fault_vout_ov_warn_cnt	7000_3C00_H 7000_4000_H	[3:2]	Output overvoltage warning count. Defines the number of consecutive 50 MHz samples by which the output voltage must exceed the warning threshold in order to assert a warning. 0 = 1 sample, 1 = 2 samples 2 = 4 samples, 3 = 8 samples
fault_vout_uv_fault_cnt	7000_3C00_H 7000_4000_H	[5:4]	Output undervoltage fault count. Defines the number of consecutive 50 MHz samples by which the output voltage must exceed the fault threshold in order to assert a fault. 0 = 1 sample, 1 = 2 samples 2 = 4 samples, 3 = 8 samples
fault_vout_uv_warn_cnt	7000_3C00_H 7000_4000_H	[7:6]	Output undervoltage warning count. Defines the number of consecutive 50 MHz samples by which the output voltage must exceed the warning threshold in order to assert a warning.

Name	Address (loop 0/1)	Bits	Description
			0 = 1 sample, 1 = 2 samples 2 = 4 samples, 3 = 8 samples
fault_vout_fault_hyst	7000_3C08$_H$ 7000_4008$_H$	[4:0]	Output voltage (V_{OUT}) fault (warning) de-assertion hysteresis referenced to the VSEN input (i.e., after scaling by VOUT_SCALE_LOOP). Internally, the V_{OUT} OV fault is asserted when V_{OUT} exceeds VOUT_OV_FAULT_LIMIT for fault_vout_ov_fault_cnt samples without dropping below (VOUT_OV_FAULT_LIMIT-fault_vout_fault_hyst). The samples above VOUT_OV_FAULT_LIMIT need not be consecutive but a single sample below (VOUT_OV_FAULT_LIMIT-fault_vout_fault_hyst) will reset the count. For the typical case this parameter should be set to a positive voltage. This hysteresis parameter applies to all V_{OUT} faults and warnings as shown in Table 80. LSB = 5 mV, range = -80 to 75 mV at the sense pin

The V_{OUT} fault registers include fault count and fault hysteresis. The fault count register defines the number of consecutive samples at a 50 MHz sample rate. The V_{OUT} fault comparator is placed before the V_{OUT} telemetry LPF (**tlm_kfp_vout**) to achieve fast response. The fault hysteresis works with the V_{OUT} warning and fault limit as in Table 80. The V_{OUT} fault hysteresis must be set larger than the ripple at the VSEN pin, otherwise the fault comparator would create continuous fault IRQ's when the signal is at the warning or fault threshold. Since the Vout ripple frequency is twice of the switching frequency, the high frequency and persistent fault IRQ would cause the FW run out of space to report the fault and eventually have hard reset.

Table 80 **V_{OUT} OV/UV hysteresis**

Fault	Asserted	De-asserted
VOUT_OV_FAULT	> VOUT_OV_FAULT_LIMIT	<= (VOUT_OV_FAULT_LIMIT-fault_vout_fault_hyst)
VOUT_OV_WARN	> VOUT_OV_WARN_LIMIT	<= (VOUT_OV_WARN_LIMIT-fault_vout_fault_hyst)
VOUT_UV_WARN	< VOUT_UV_WARN_LIMIT	>= (VOUT_UV_WARN_LIMIT+fault_vout_fault_hyst)
VOUT_UV_FAULT	< VOUT_UV_FAULT_LIMIT	>= (VOUT_UV_FAULT_LIMIT+fault_vout_fault_hyst)

The **fault_vout_fault_hyst** is V_{OUT} fault hysteresis after scaling by VOUT_SCALE_LOOP. For example, a 12 V output converter, VOUT_SCALE_LOOP = 0.1, the voltage ripple is 150 mV at VOUT and the ripple at VSEN pin is about 15 mV. But remember the switching noise would not be proportional scaled down at the VSEN pin thus the hysteresis must consider additional buffer for switching noise.

Figure 163 demonstrates the function of **vout_fault_hysteresis** and **vout_ov_warn_cnt**, **vout_ov_fault_cnt**.

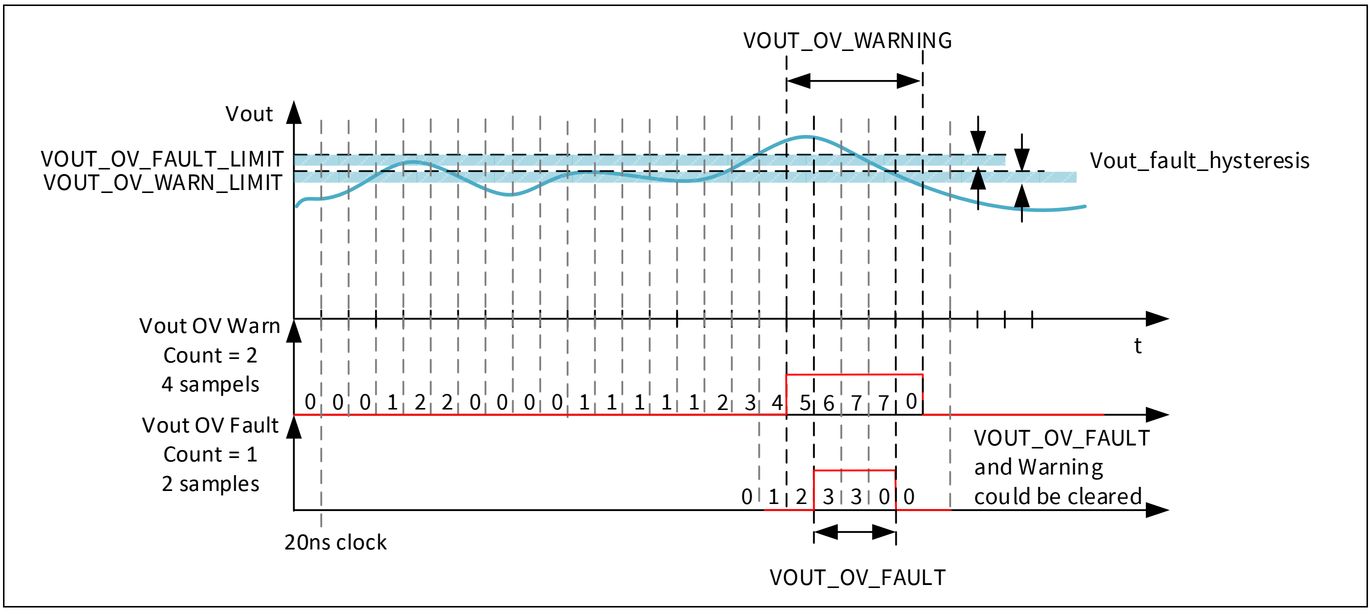


Figure 163 V_{OUT} OV fault hysteresis and fault count

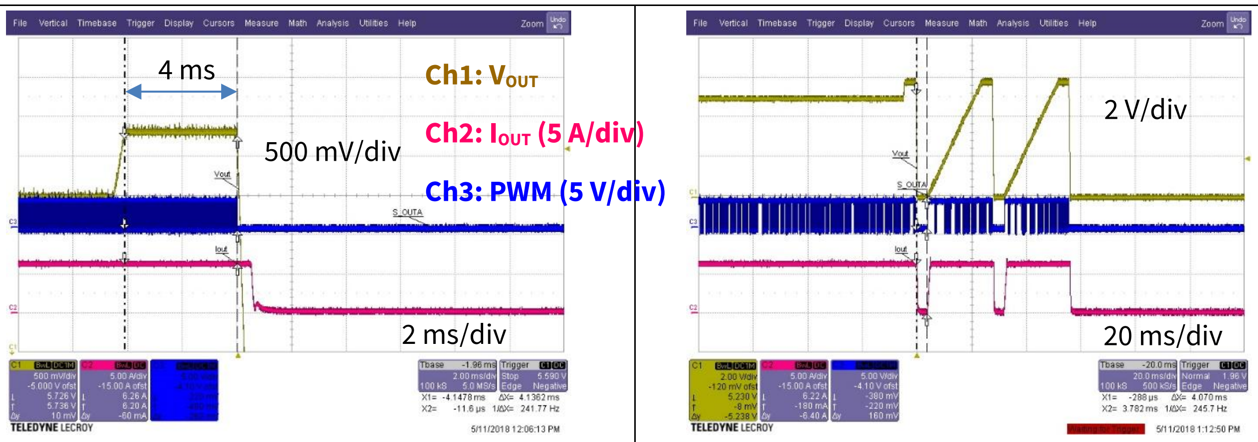
12.3.3 V_{OUT} OVP/UVP fault response

Output overvoltage and undervoltage fault response can be configured as in Table 81.

Table 81 **Voltage, temperature and TON_MAX fault responses**

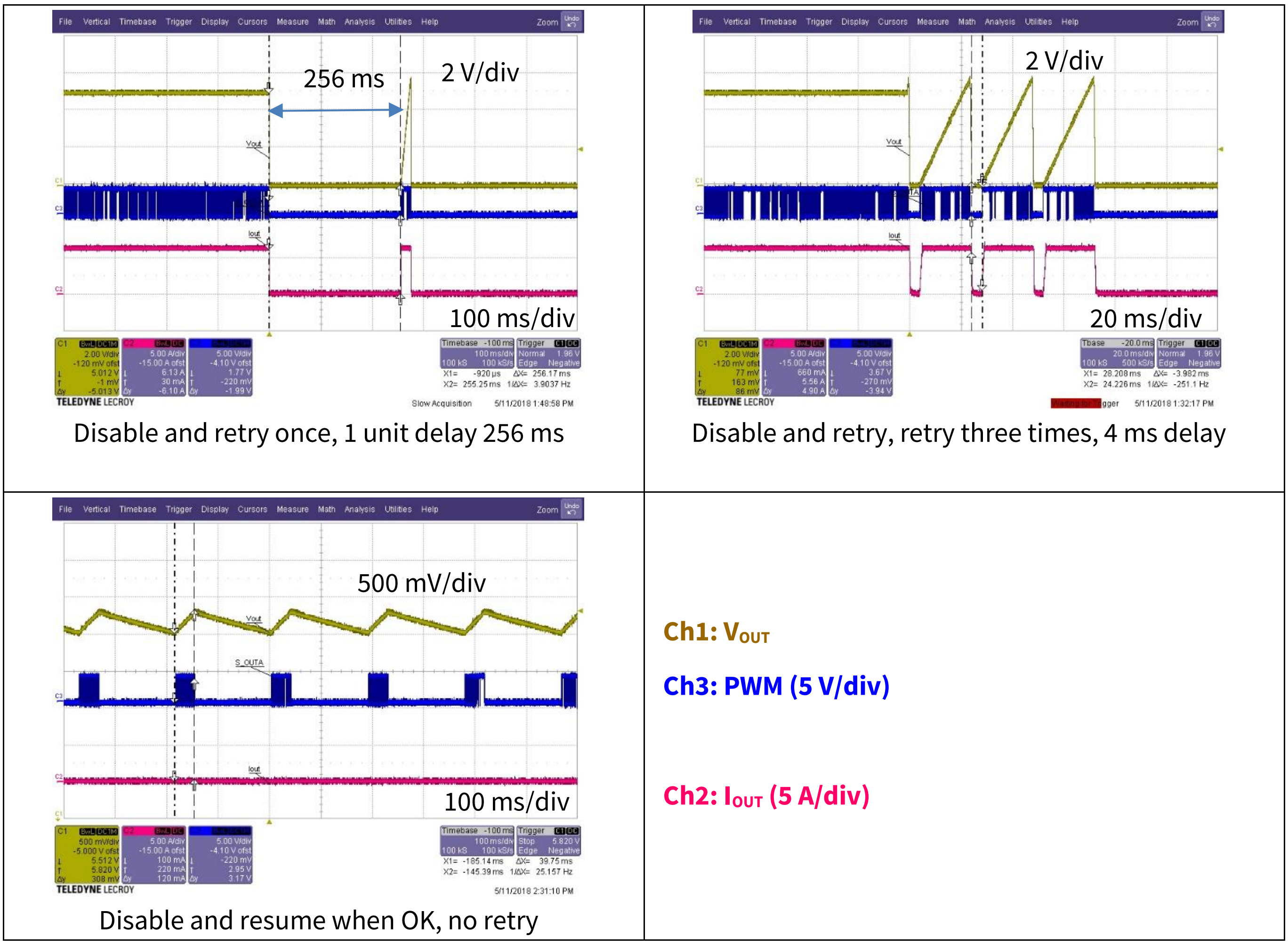
Response	Description
Ignore	Continues operation without interruption.
Operate for delay time	Continues operation for the delay time. If the fault condition is still present at the end of the delay time, the unit responds as programmed in the retry setting.
Disable and retry	Shuts down and responds according to the retry setting.
Disable and resume when OK	Shuts down while the fault is present. The output remains disabled until the fault is cleared.

Figure 164 shows examples of the V_{OUT} OV fault response of a HBCT converter with 5 V output.



Operation for delay time, no retry, 4 ms delay

Operate for delay time, 2 retries, 4 ms delay



Disable and retry once, 1 unit delay 256 ms

Disable and retry, retry three times, 4 ms delay

Disable and resume when OK, no retry

Ch1: V_{OUT}

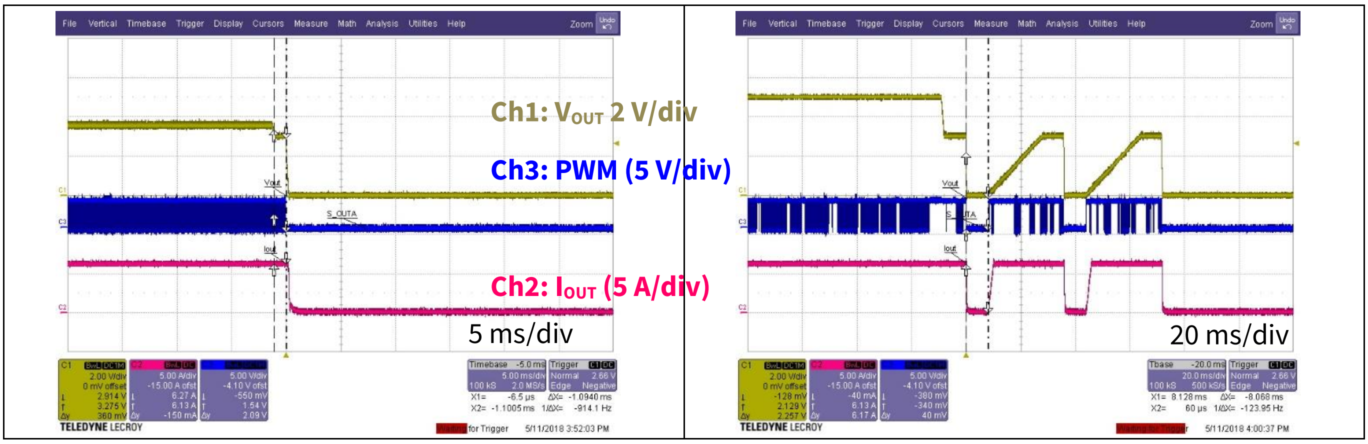
Ch3: PWM (5 V/div)

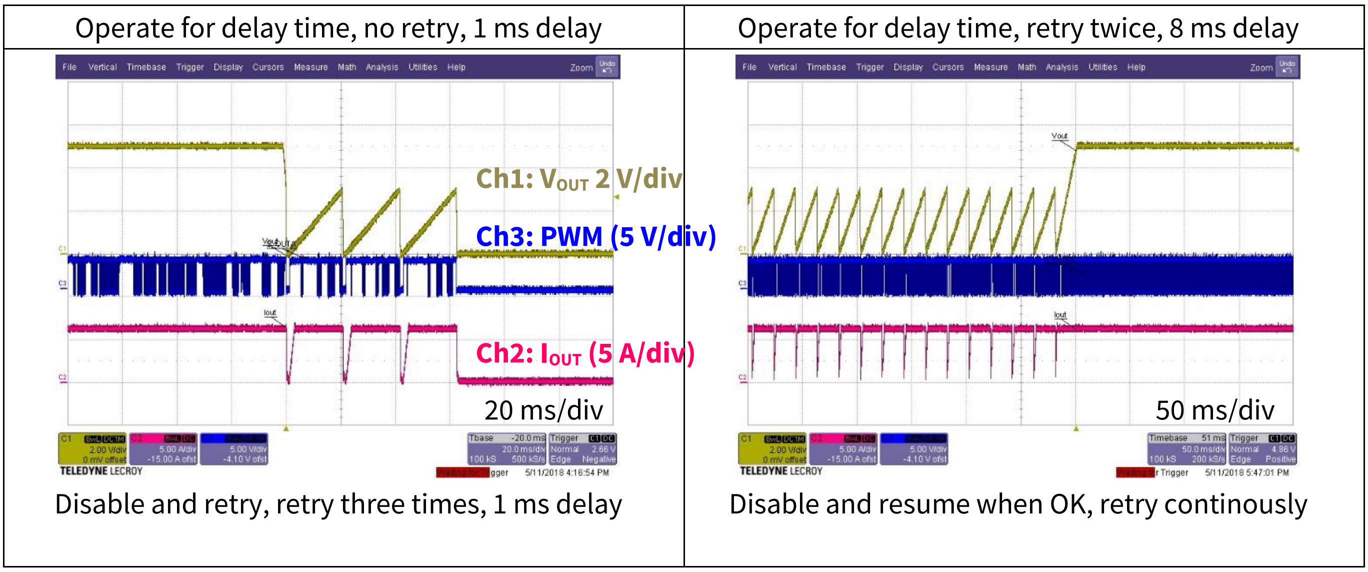
Ch2: I_{OUT} (5 A/div)

Figure 164 Output OVP waveform (V_{OUT} = 5 V, OVP fault threshold = 5.8 V, OVP warning threshold = 5.5 V)

When the V_{OUT} OV fault response is set to "disable and resume when OK", the converter shuts down the output when V_{OUT} is higher than VOUT_OV_FAULT_LIMIT. It resumes switching when V_{OUT} is lower than VOUT_OV_WARN_LIMIT. VOUT_OV_WARN_LIMIT is used as the hysteresis threshold of VOUT_OV_FAULT. This hysteresis should not be confused with the fault_vout_fault_hyst, which defines the HW hysteresis of the V_{OUT} OV and UV comparators. The PMBus command VOUT_OV_WARN_LIMIT defines the FW fault hysteresis for VOUT_OV_FAULT_RESPONSE.

Figure 165 shows examples of V_{OUT} UV fault response.




Operate for delay time, no retry, 1 ms delay

Operate for delay time, retry twice, 8 ms delay

Disable and retry, retry three times, 1 ms delay

Disable and resume when OK, retry continously

Figure 165 V_{OUT} UVP response (V_{OUT} = 5 V, VOUT_UV_FAULT_LIMIT = 3.199 V, VOUT_MIN = 0.5 V)

12.3.4 V_{OUT} max/min warning

Table 82 V_{OUT} max/min PMBus command descriptions

Command name	Command code	Description
VOUT_MAX	24_H	This command is used to limit the maximum output voltage. Target voltages (including droop) above this limit will be clamped to this level. Data format = VOUT_MODE
VOUT_MIN	$2B_H$	This command is used to limit the minimum output voltage. Target voltages (including droop) below this limit will be clamped to this level. Data format = VOUT_MODE

The VOUT_MAX and VOUT_MIN commands provide a safeguard against a user accidentally setting the output voltage to a possibly destructive level rather than being the primary output OV or UV protection. If the output is set to a voltage exceeding the VOUT_MAX and VOUT_MIN bounds, the output voltage will be clamped to VOUT_MAX or VOUT_MIN, and the VOUT_MAX_MIN_WARNING will flag in STATUS_VOUT.

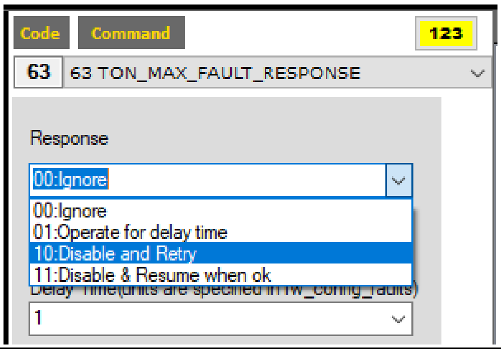
12.3.5 TON MAX fault

Table 83 TON MAX fault PMBus command descriptions

Command name	Command code	Description
TON_MAX_FAULT_LIMIT	62_H	This command sets an upper limit, in milliseconds, on how long the unit can attempt to power up the output without reaching the output undervoltage fault limit. A setting of 0 means that there is no limit and the unit can attempt to bring up the output voltage indefinitely. Data format = LINEAR11, U8.2, LSB 0.25 ms, range 0 to 255 ms Data format = LINEAR11, U9.1, LSB 0.5 ms, range 0 to 511 ms

Command name	Command code	Description
		Data format = LINEAR11, U10.0, LSB 1 ms, range 0 to 1023 ms
TON_MAX_FAULT_RESPONSE	63_H	Defines what action to take in response to a TON_MAX fault.

The TON_MAX_FAULT could happen in an overload start-up condition. The output current hits the IOUT_OC_LIMIT and the ramp-up of output voltage is slowed down due to over-current clamping when the IOUT OC response is set to the constant current limit. The TON_MAX_FAULT will assert at the end of the TON_MAX_FAULT_LIMIT time if the output voltage is lower than the VOUT_UV_FAULT_LIMIT. The fault response is set by the TON_MAX_FAULT_RESPONSE command, as shown in **Figure 166**.



63 | 63 TON_MAX_FAULT_RESPONSE

Response

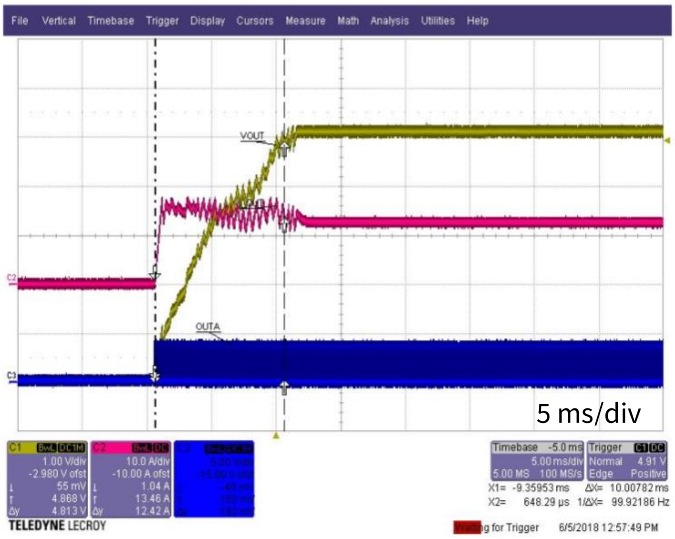
00:Ignore

00:Ignore
01:Operate for delay time
10:Disable and Retry
11:Disable & Resume when ok

Delay Time(units are specified in fw_config_faults)

1

Figure 166 TON_MAX_FAULT_RESPONSE



HBCT converter rated 5 V/25 A, start-up at 25 A load with 10 mF extended output capacitor.

I_{OUT} OC response set to constant current.

TON_RISE = 3 ms, TON_MAX_FAULT_LIMIT = 4 ms

VOUT_UV_FAULT_LIMIT = 4.5 V

Ignore TON_MAX_FAULT.

V_{OUT} reached target voltage at 12 ms due to constant current limit. By the end of TON_MAX_FAULT_LIMIT (4 ms), V_{OUT} was lower than 4.5 V and TON_MAX_FAULT was asserted.

Ch1: V_{OUT}, 1 V/div

Ch2: I_{OUT}, 20 A/div

Ch3: PWM output

Figure 167 TON_MAX_FAULT waveforms

12.3.6 TOFF_MAX warning

The TOFF_MAX_WARN_LIMIT defines the maximum V_{OUT} fall time under a controlled soft turn-off condition. It does not apply to the hard-stop turn-off. Unlike the TON_MAX_FAULT, the TOFF_MAX is just a warning status and does not provide response options. Once the TOFF_MAX_WARN_LIMIT is triggered, bit [1] of the status command STATUS_VOUT will be flagged.

Table 84 TOFF_MAX warning PMBus command descriptions

Command name	Command code	Description
TOFF_MAX_WARN_LIMIT	66_H	This command sets an upper limit, in milliseconds, on how long the unit can attempt to power down the output without reaching 12.5 percent of the output voltage programmed at the time the unit is turned off. A setting of 0 means that there is no limit and that the unit waits indefinitely for the output voltage to decay. Data format = LINEAR11, U8.2, LSB = 0.25 ms, range = 0 to 255 ms Data format = LINEAR11, U9.1, LSB = 0.5 ms, range = 0 to 511 ms Data format = LINEAR11, U10.0, LSB = 1 ms, range = 0 to 1023 ms

12.3.7 Power good

Table 85 Power good PMBus command descriptions

Command name	Command code	Description
POWER_GOOD_ON	$5E_H$	This command sets the output voltage at which a POWER_GOOD signal should be asserted, indicating that the output voltage is valid. The power good polarity can be set either active high or active low. Data format = VOUT_MODE
POWER_GOOD_OFF	$5F_H$	This command sets the output voltage at which an optional POWER_GOOD signal should be de-asserted, indicating that the output voltage is not valid. Data format = VOUT_MODE

If the converter fails to reach the POWER_GOOD_ON threshold after start-up, a PG_STATUS fault will be reported, signaling that the output power is not good.

Power good test example:

VOUT_COMMAND = 3.3 V. POWER_GOOD_ON = 3 V, POWER_GOOD_OFF = 2.8 V. Power good logic = active low.

In **Figure 168**, power good is asserted when V_{OUT} is higher than 3 V at start-up, and is de-asserted when V_{OUT} is lower than 2.8 V at soft turn-off. The power good logic is configured as active low, thus the signal is low when output voltage is in good condition.

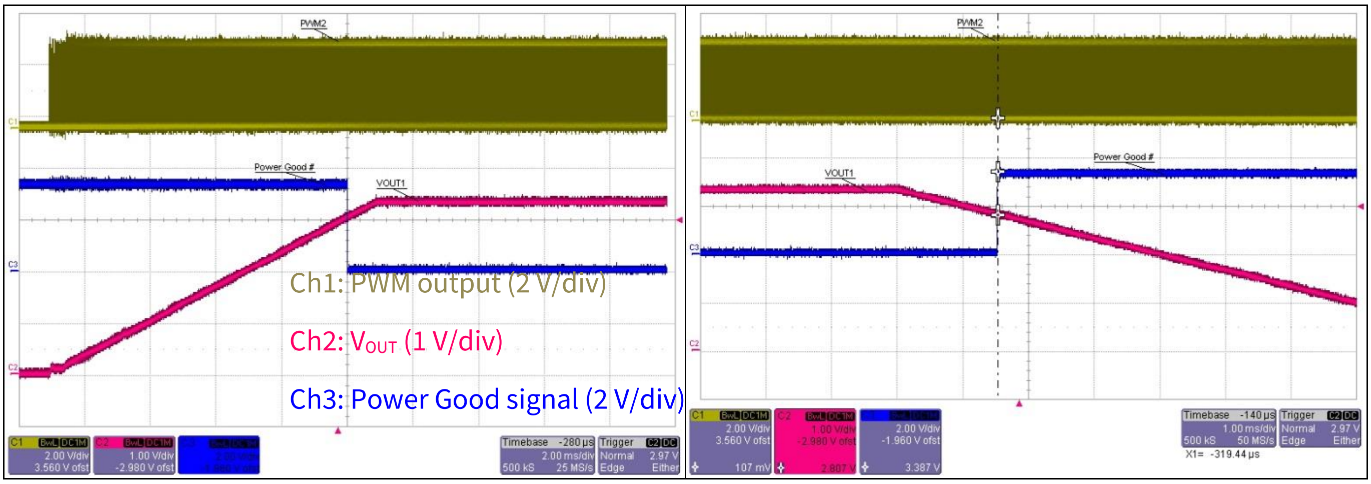


Figure 168 **Power good (active low) at soft-start and soft-off**

Please note, in a hard-stop turn-off situation, power good is immediately de-asserted, ignoring output voltage level.

12.4 Output OCP/UCP

12.4.1 I_{OUT} OCP/UCP PMBus commands

Table 86 describes the PMBus commands that are relevant to the I_{OUT} OCP and UCP.

Table 86 **I_{OUT} OCP/UCP relevant PMBus command descriptions**

Command name	Command code	Description
IOUT_OC_FAULT_LIMIT	46_H	This command sets the output over-current fault threshold. Data format = LINEAR 11, U8.0 LSB = 1 A, range = 0 to 255 A
IOUT_OC_FAULT_RESPONSE	47_H	This command instructs the device on what action to take in response to an output over-current fault. See section 12.4.4 for I_{OUT} OCP fault response.
IOUT_OC_LV_FAULT_LIMIT	48_H	In the cases where the response to an over-current condition is to operate in CCM by dropping the output voltage, this threshold defines the minimum voltage below which the output will shut down. Data format = VOUT_MODE
IOUT_OC_WARN_LIMIT	$4A_H$	This command sets the output over-current warning threshold. This value is typically less than the output over-current fault threshold. It is used as the hysteresis threshold for IOUT_OC_FAULT. Data format = LINEAR 11, U8.0 LSB = 1 A, range = 0 to 255 A
IOUT_UC_FAULT_LIMIT	$4B_H$	This command sets the output under-current fault threshold. Data format = LINEAR 11, S8.0 LSB = 1 A, range = -128 to 127 A

Command name	Command code	Description
IOUT_UC_FAULT_RESPONSE	$4C_H$	This command instructs the device on what action to take in response to an output under-current fault. See section 12.4.5 for I_{OUT} UCP fault response.
MFR_IOUT_APC	EA_H	This command sets the current sense gain. Data format = LINEAR 11, U1.9 LSB = 0.00195A, range = 0 to 1.998 A Data format = LINEAR 11, U2.8 LSB = 0.0039 A, range 0 to 3.996 A

12.4.2 I_{OUT} OC fast protection

Besides the regular output OCP, the XDPP1100 also offers a fast over-current protection. The difference between the two faults is that the regular over-current is sensed after a LPF, and the fast OCP is sensed before the LPF. The LPF is defined by **tlm_kfp_iout** in Table 88.

Table 87 I_{OUT} fast OCP PMBus command descriptions

Command name	Command code	Description
MFR_IOUT_OC_FAST_ FAULT_RESPONSE	CA_H	This command instructs the device on what action to take in response to a fast output over-current fault (exceeding MFR_IOUT_OC_FAST_FAULT_LIMIT).
MFR_IOUT_OC_FAST_ FAULT_LIMIT	$D1_H$	This command sets the fast output over-current fault threshold. Data format = LINEAR11, U8.0 LSB = 1 A, range = 0 to 255 A

12.4.3 I_{OUT} OCP/UCP registers

Table 88 describes the registers that are relevant to the I_{OUT} OCP and UCP.

Table 88 I_{OUT} OC/UC relevant register descriptions

Name	Address (loop 0/1)	Bits	Description
Fault peripheral			

fault_iout_oc_fault_cnt	7000_3C00_H
	7000_4000_H
		[12:8]	Output over-current fault count. Defines the number of consecutive switching cycles (T_{sw}) by which the low-pass filtered current must exceed the fault threshold in order to assert a fault.
			Count = fault_iout_oc_fault_cnt + 1
			Range = 1 to 32 switching cycles

fault_iout_oc_warn_cnt	7000_3C00_H 7000_4000_H	[17:13]	Output over-current warning count. Defines the number of consecutive switching cycles (T_{sw}) by which the low-pass filtered current must exceed the warning threshold

Name	Address (loop 0/1)	Bits	Description

			in order to assert a warning.
			Count = fault_iout_oc_warn_cnt + 1
			Range = 1 to 32 switching cycles

fault_mfr_iout_oc_fast_cnt	7000_3C00$_H$
	7000_4000$_H$
		[22:18]	MFR_IOUT_OC_FAST over-current fault count. Defines the number of consecutive switching cycles (T_{sw}) by which the cycle averaged current must exceed the fault threshold in order to assert a fault.
			Count = fault_mfr_iout_oc_fast_cnt + 1
			Range = 1 to 32 switching cycles

fault_iout_uc_fault_cnt	7000_3C00$_H$
	7000_4000$_H$
		[27:23]	Output under-current warning count. Defines the number of consecutive switching cycles (T_{sw}) by which the cycle averaged current must exceed the fault threshold in order to assert a fault.
			Count = fault_iout_uc_fault_cnt + 1
			Range = 1 to 32 switching cycles

fault_iout_oc_fault_hyst	7000_3C08$_H$ 7000_4008$_H$	[15:10]	Output over-current (IOUT_OC) fault (warning) de-assertion hysteresis. Internally, the I_{OUT} OC fault is asserted when I_{OUT} exceeds IOUT_OC_FAULT_LIMIT for fault_iout_oc_fault_cnt samples without dropping below (IOUT_OC_FAULT_LIMIT-fault_iout_oc_fault_hyst). The samples above IOUT_OC_FAULT_LIMIT need not be consecutive but a single sample below (IOUT_OC_FAULT_LIMIT-fault_iout_oc_fault_hyst) will reset the count. For the typical case this parameter should be set to a positive current. This hysteresis parameter applies to all I_{OUT} OC faults and warnings, as shown in Table 89. Resolution = 0.25 A, range = -8 to 7.75 A
fault_iout_uc_fault_hyst	7000_3C08$_H$ 7000_4008$_H$	[21:16]	Output under-current (IOUT_UC) fault de-assertion hysteresis. Internally, the I_{OUT} UC fault is asserted when I_{OUT} exceeds IOUT_UC_FAULT_LIMIT for fault_iout_uc_fault_cnt samples without rising above (IOUT_UC_FAULT_LIMIT+fault_iout_uc_fault_hyst). The samples below IOUT_UC_FAULT_LIMIT need not be consecutive but a single sample above (IOUT_UC_FAULT_LIMIT+fault_iout_uc_fault_hyst) will reset the count. For the typical case this parameter should be set to a positive current. Resolution = 0.25 A, range = -8 to 7.75 A
Telemetry peripheral			
tlm_kfp_iout	7000_3400$_H$ 7000_3800$_H$	[27:22]	Output current telemetry LPF coefficient index. Note that exponent settings greater than 9 are clamped to 9. Set to 63 to bypass filter.

Name	Address (loop 0/1)	Bits	Description
			• kfp_exp = tlm_kfp_iout [5:2] • kfp_man = 4 + tlm_kfp_iout [1:0] • kfp = kfp_man * 2^(kfp_exp – 13) • F3db (kHz) = [kfp/(1-kfp)] * F_{switch} (kHz)/2π

All the telemetry LPF can be configured in the design tool "Basic Configuration" and "Telemetry" tab, as shown in **Figure 30**. The tool helps the user to calculate the cut-off frequency of the LPF based on the register value and the switching frequency.

The output fast over-current fault protection is sensed before the LPF, thus it is suggested to set the **fault_mfr_iout_oc_fast_cnt** to a larger value (i.e. 8) to avoid triggering the fault by ripple or noise.

Table 89 I_{OUT} OC and UC hysteresis

Fault	Asserted	De-asserted
IOUT_OC_FAULT	> IOUT_OC_FAULT_LIMIT	<= (IOUT_OC_FAULT_LIMIT-fault_iout_oc_fault_hyst)
IOUT_OC_WARN	> IOUT_OC_WARN_LIMIT	<= (IOUT_OC_WARN_LIMIT-fault_iout_oc_fault_hyst)
IOUT_OC_FAST_FAULT	> MFR_IOUT_OC_FAST_FAULT_LIMIT	<= (MFR_IOUT_OC_FAST_FAULT_LIMIT-fault_iout_oc_fault_hyst)
IOUT_UC_FAULT	< IOUT_UC_FAULT_LIMIT	>= (IOUT_UC_FAULT_LIMIT+fault_iout_uc_fault_hyst)

Figure 169 demonstrates the function of **iout_oc_fault_hysteresis** and **iout_oc_warn_cnt**, **iout_oc_fault_cnt**. In this example, **iout_oc_warn_cnt** = 3, **iout_oc_fault_cnt** = 3, which defines the number of consecutive samples over the current limit must be 4 to assert fault.

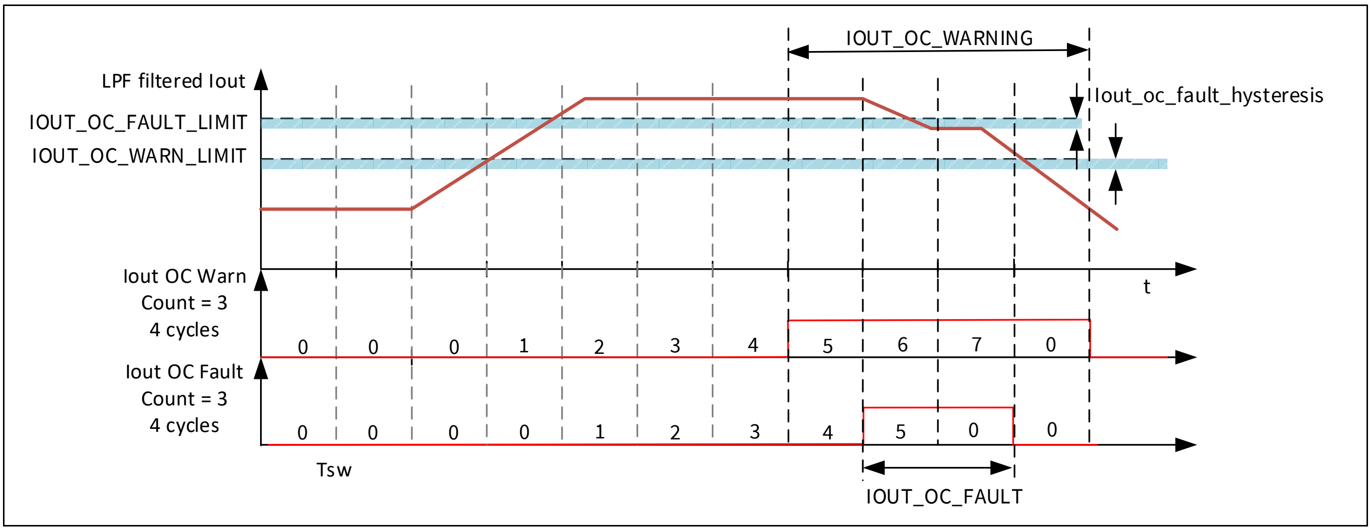


Figure 169 I_{OUT} OC fault hysteresis and fault count

12.4.4 I_{OUT} OC fault response

Output over-current fault response can be configured to be one of the following, as in Table 90. The output fast over-current fault response can be configured as in Table 91.

Table 90 I_{OUT} **OC fault response**

Response	Description
Continues to operate	Continues to operate indefinitely while maintaining the output current at the value set by IOUT_OC_FAULT_LIMIT without regard to the output voltage (CC).
Continues to operate with OC LV	Continues to operate indefinitely while maintaining the output current at the value set by IOUT_OC_FAULT_LIMIT as long as the output voltage remains above the minimum value specified by IOUT_OC_LV_FAULT_LIMIT. If the output voltage is pulled down to less than that value, then the PMBus device shuts down and responds according to the retry setting.
Continues to operate for delay time	Continues operation for delay time, maintaining the output current that set by IOUT_OC_FAULT_LIMIT. If the device is still operating in current limiting at the end of the delay time, the device responds as programmed by the retry setting.
Shut down and retry	Shuts down and responds according to the retry setting.

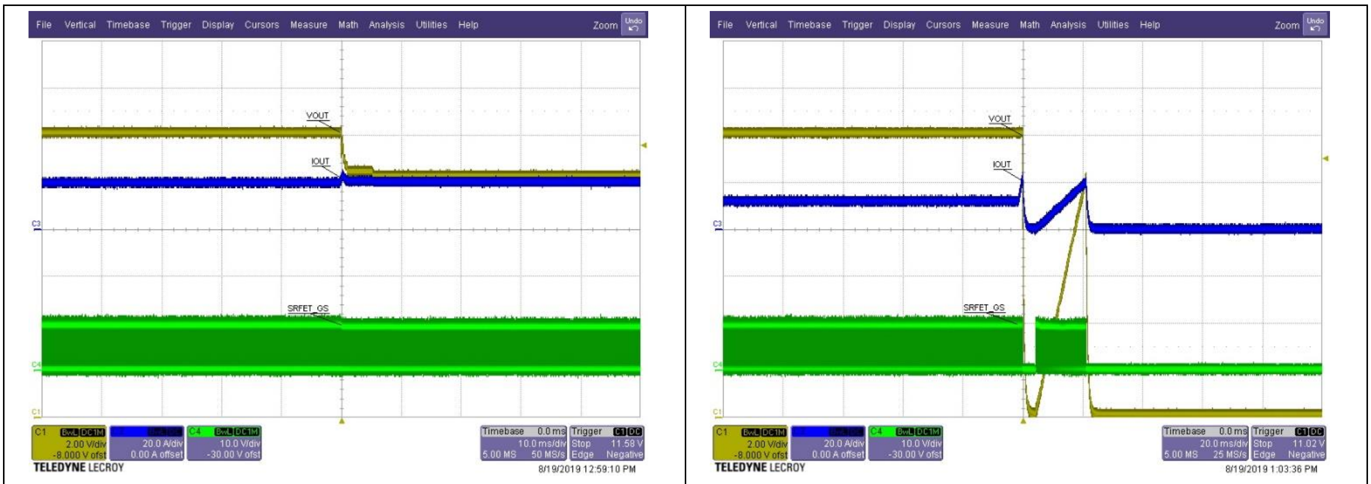
Table 91 I_{OUT} **fast OC fault response**

Response	Description
Continues to operate	Continues operation with CC limit.
Shutdown and retry	Shuts down and responds according to the retry setting.

Figure 170 shows output over-current fault waveforms tested with HBCT 12 V/20 A converter. To verify the functionality of the OC fault protection, the fault thresholds are set to IOUT_OC_FAULT_LIMIT = 22 A, IOUT_OC_WARN_LIMIT = 21 A, IOUT_OC_LV_FAULT= 4.599 V.

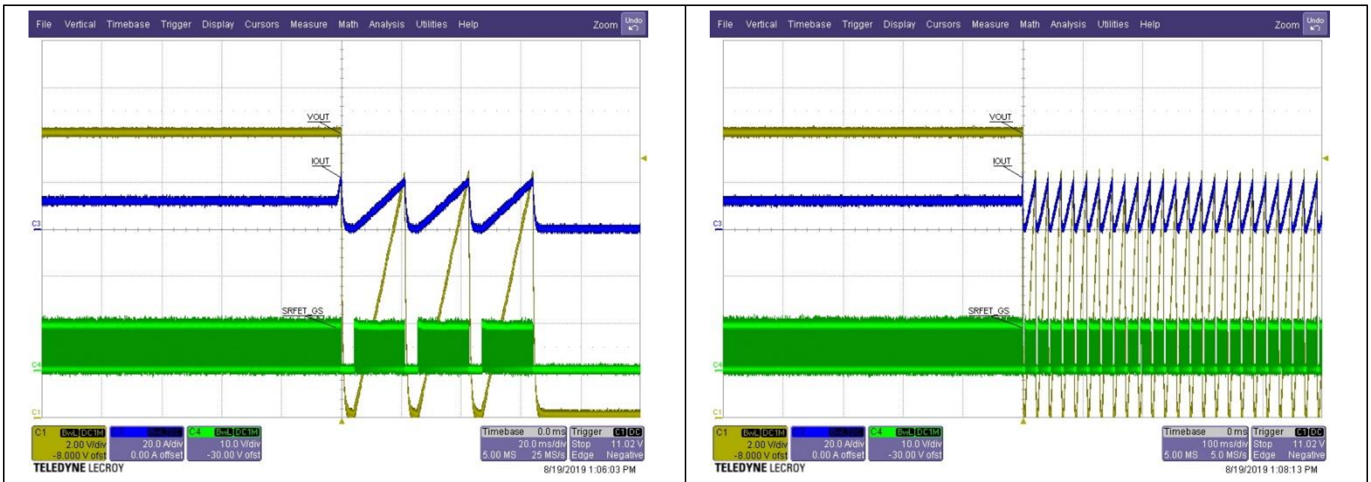
The electronic load is set to constant resistor mode to test constant current operation.

Ch1 = V_{OUT} (2 V/div), Ch3 = I_{OUT} (20 A/div), Ch4 = secondary V_{gs} (10 V/div)



48 V input, 20 A to over-current, CC
(a)

48 V input, 12 A to over-current, shut down with one retry (b)



48 V input, 12 A to over-current, shut down and retry three times (c)

48 V input, 12 A to over-current, shut down and retry with continous retry (d)

Figure 170 Output OCP waveform

12.4.5 I_{OUT} UC fault response

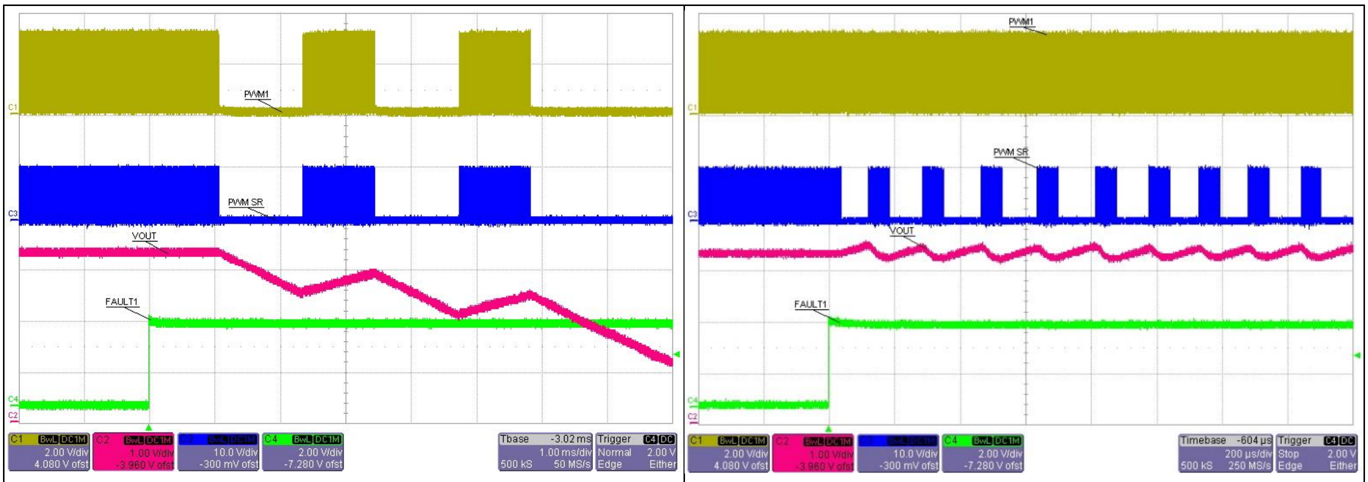
Output under-current fault response can be configured to be one of the following, as in Table 92.

Table 92 I_{OUT} UC fault response

Response	Description
Ignore	Continues operation without interruption.
Operate for delay time	Continues operation for delay time. If the fault condition is still present at the end of the delay time, the unit responds as programmed retry setting.
Disable and retry	Shuts down SR and responds according to the retry setting.
Disable and resume when OK	Shuts down SR while the fault is present.

Figure 171 shows I_{OUT} UC fault waveforms tested with ACF 3.3 V/30 A converter. To verify the functionality of the UC fault protection, the fault thresholds were set to IOUT_UC_FAULT_LIMIT = 1 A, and tested at no load. Please note, the “disable operation” at under-current fault only disables SR gate drive and won’t stop primary PWM.

Ch1: primary PWM (2 V/div), Ch2: V_{OUT} (1 V/div), Ch3: SR gate (10 V/div), Ch4: fault signal



Operate for delay time, 1 ms delay, retry twice

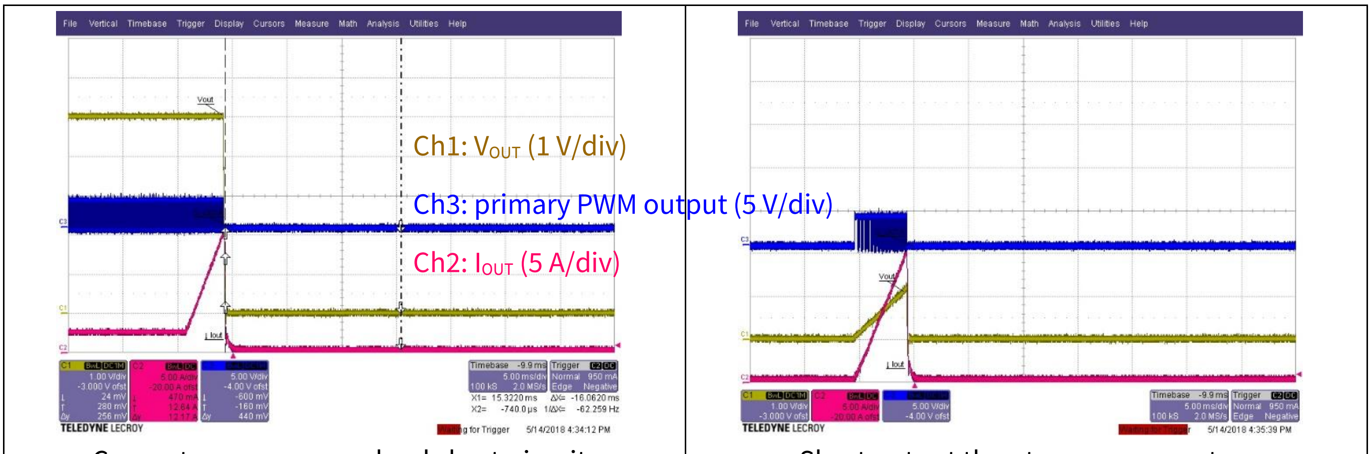
Disable and resume when OK (only disable SR)

Figure 171 Output UCP waveform

12.4.6 Short circuit protection

Output short circuit protection can be achieved by fast OCP or by SCP. **Figure 172** shows short circuit protection by fast over-current fault.

The protection was tested with a 5 V/25 A HBCT converter to see that the short-circuit was applied during operation and during start-up. The Iout_OC_fast_fault was reported. MFR_IOUT_OC_FAST_RESPONSE was set to shut down no retry, **fault_mfr_iout_oc_fast_cnt** =8.



Converter response on load short-circuit

Short output then turn-on converter

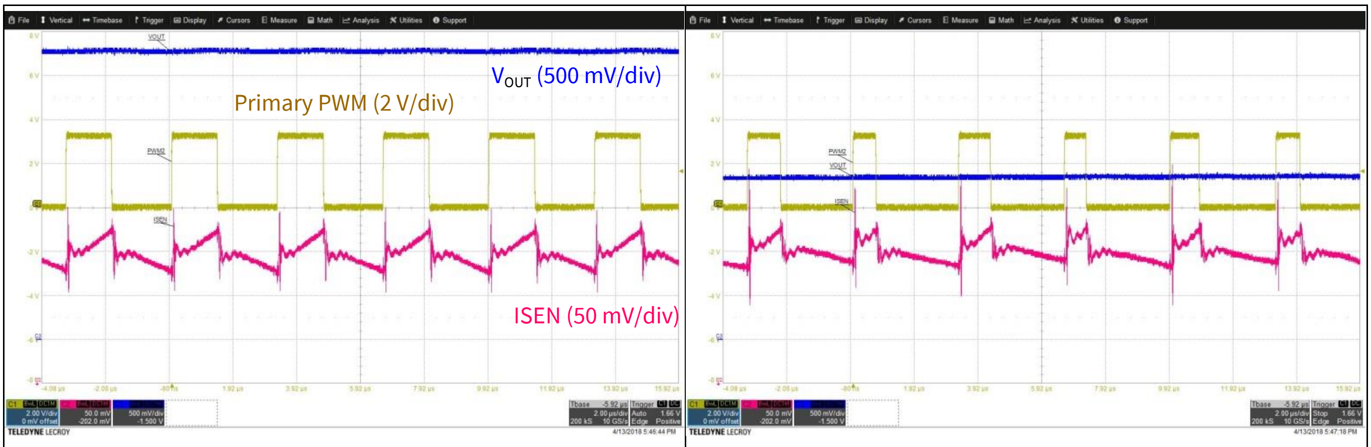
Figure 172 Output short-circuit waveform (48 V input)

12.4.7 Cycle-by-cycle PCL

Register **ce_current_limit** sets the cycle-by-cycle PCL threshold. The **ce0_current_limit** sets the limit for ISEN input, and the **ce1_current_limit** sets the limit for BISEN input. While the PCL is enabled, the controller truncates the PWM duty-cycle when the sensed current exceeds the limit. Setting the register to 0 will disable the function. The PCL should be calculated by the following equation:

$$\text{ce\_current\_limit} = \frac{Peak\_Current}{MFR\_IOUT\_APC} \tag{11.1}$$

The Peak_Current must be set higher than the maximum rated output current, plus the inductor ripple current. For example, an ACF converter output is 3.3 V/30 A, and the ripple current of the board is 6.5 A at 48 V input. The Peak_Current must be set higher than (30 + 6.5/2) = 33.25 A at 48 V input.



48 V, 30 A, no PCL

48 V, 30 A, with cycle-by-cycle PCL

Figure 173 Output PCL waveform (48 V input)

Figure 173 shows the PCL waveform of the 3.3 V/30 A ACF converter. The **ce0_current_limit** = 110, IOUT_APC = 0.32 A. The PCL threshold is 35 A. With the cycle-by-cycle PCL, the primary PWM pulse was truncated and the output voltage folded back.

12.4.8 Current share fault

A current share fault will be asserted if the controller enables the active current sharing and the current sharing was not able to correct the error within the MFR_ISHARE_THRESHOLD limit. Usually the output voltage adjusts to the active current sharing clamps (**ishare_clamp_pos**, **ishare_clamp_neg**), but there may still be a current error larger than the MFR_ISHARE_THRESHOLD or the **ishare_dead_zone**. The “current share fault” is reported and bit 3 of the STATUS_IOUT is set to 1. More information on active current sharing can be found in chapter 5.

12.4.9 Output over-power warning

The PMBus command POUT_OP_WARN_LIMIT sets the output power warning threshold. If the output power is higher than the limit, the POUT_OP_WARN will be asserted and bit 0 of the STATUS_IOUT is set to 1.

The maximum output power limit could be set to 4096 W when exponent 2 is selected. If the exponent is 0, the maximum output power limit can be set to 1024 W.

12.5 Input OVP/UVP

Table 93 describes the PMBus commands that are relevant to the V_{IN} OVP and UVP. The maximum input voltage can be configured to 127.75 V. To use the XDPP1100 in high-voltage applications, such as 400 V input voltage, a FW patch is required to shift the LSB resolution and scale up the input voltage range.

12.5.1 V_{IN} OVP/UVP PMBus commands

Table 93 V_{IN} OVP/UVP relevant PMBus command descriptions

Command name	Command code	Description
VIN_OV_FAULT_LIMIT	55_H	This command sets the input overvoltage fault threshold. Data format = LINEAR 11, U7.2 LSB = 0.25 V, range = 0 to 127.75 V
VIN_OV_FAULT_RESPONSE	56_H	This command instructs the device on what action to take in response to an input overvoltage fault. See section 12.5.3 for V_{IN} OVP fault response.
VIN_OV_WARN_LIMIT	57_H	This command sets the input overvoltage warning threshold. This value is typically less than the input overvoltage fault threshold. It is used as the hysteresis threshold for VIN_OV_FAULT. Data format = LINEAR 11, U7.2 LSB = 0.25 V, range = 0 to 127.75 V
VIN_UV_WARN_LIMIT	58_H	This command sets the input undervoltage warning threshold. This value is typically higher than the input undervoltage fault threshold. It is used as the hysteresis threshold for VIN_UV_FAULT. Data format = LINEAR 11, U6.2 LSB = 0.25 V, range = 0 to 63.75 V

Command name	Command code	Description
VIN_UV_FAULT_LIMIT	59_H	This command sets the input undervoltage fault threshold. Data format = LINEAR 11, U6.2 LSB = 0.25 V, range = 0 to 63.75 V
VOUT_UV_FAULT_RESPONSE	$5A_H$	This command instructs the device on what action to take in response to an input undervoltage fault. See section 12.5.3 for V V_{IN} in UVP fault response.

12.5.2 V_{IN} OVP/UVP registers

Table 94 describes the registers relevant to the V_{IN} OVP and UVP.

Table 94 V_{OUT} OV/UV relevant register descriptions

Name	Address (loop 0/1)	Bits	Description
Fault peripheral			
fault_vin_ov_fault_cnt	7000_3C04_H 7000_4004_H	[1:0]	Input overvoltage fault count. Defines the number of consecutive switching cycles (T_{sw}) by which the input voltage must exceed the fault threshold in order to assert a fault. 0 = 1 T_{sw}, 1 = 2 T_{sw}, 2 = 4 T_{sw}, 3 = 8 T_{sw}
fault_vin_ov_warn_cnt	7000_3C04_H 7000_4004_H	[3:2]	Input overvoltage warning count. Defines the number of consecutive switching cycles (T_{sw}) by which the input voltage must exceed the warning threshold in order to assert a warning 0 = 1 T_{sw}, 1 = 2 T_{sw}, 2 = 4 T_{sw}, 3 = 8 T_{sw}
fault_vin_uv_fault_cnt	7000_3C04_H 7000_4004_H	[5:4]	Input undervoltage fault count. Defines the number of consecutive switching cycles (T_{sw}) by which the input voltage must exceed the fault threshold in order to assert a fault. 0 = 1 T_{sw}, 1 = 2 T_{sw}, 2 = 4 T_{sw}, 3 = 8 T_{sw}
fault_vin_uv_warn_cnt	7000_3C04_H 7000_4004_H	[7:6]	Input undervoltage warning count. Defines the number of consecutive switching cycles (T_{sw}) by which the input voltage must exceed the warning threshold in order to assert a warning. 0 = 1 T_{sw}, 1 = 2 T_{sw}, 2 = 4 T_{sw}, 3 = 8 T_{sw}
fault_vin_fault_hyst	7000_3C04_H 7000_4004_H	[23:18]	Input voltage (V_{IN}) fault (warning) de-assertion hysteresis. Internally, the V_{IN} OV fault is asserted when V_{IN} exceeds VIN_OV_FAULT_LIMIT for fault_vin_ov_fault_cnt samples without dropping below (VIN_OV_FAULT_LIMIT-fault_vin_fault_hyst). The samples above VIN_OV_FAULT_LIMIT need not be consecutive but a single sample below (VIN_OV_FAULT_LIMIT-fault_vin_fault_hyst) will reset the count. For the typical case this parameter should be set to a positive voltage. This hysteresis parameter applies to all V_{IN} faults and warnings as shown in Table 95.

Name	Address (loop 0/1)	Bits	Description
			LSB = 0.125 V, range = -4 to 3.875 V

Table 95 V_{IN} **OV/UV hysteresis**

Fault	Asserted	De-asserted
VIN_OV_FAULT	> VIN_OV_FAULT_LIMIT	<= (VIN_OV_FAULT_LIMIT-fault_vin_fault_hyst)
VIN_OV_WARN	> VIN_OV_WARN_LIMIT	<= (VIN_OV_WARN_LIMIT-fault_vin_fault_hyst)
VIN_UV_WARN	< VIN_UV_WARN_LIMIT	>= (VIN_UV_WARN_LIMIT+fault_vin_fault_hyst)
VIN_UV_FAULT	< VIN_UV_FAULT_LIMIT	>= (VIN_UV_FAULT_LIMIT+fault_vin_fault_hyst)

For example, a 48 V input DC-DC converter has 80 V input overvoltage limit. To set the fault hysteresis to 0.5 V, the **fault_vin_fault_hyst** is calculated by:

fault_vin_fault_hyst = hysteresis_voltage/LSB = 0.5 V/0.125 V = 4

Figure 174 demonstrates the function of **vin_fault_hysteresis** and **vin_ov_warn_cnt**, **vin_ov_fault_cnt**. Please note the V_{IN} fault is detected after the telemetry filter **tlmX_kfp_vin**. To have fast V_{IN} fault response, bypass the filter by setting it to 63.

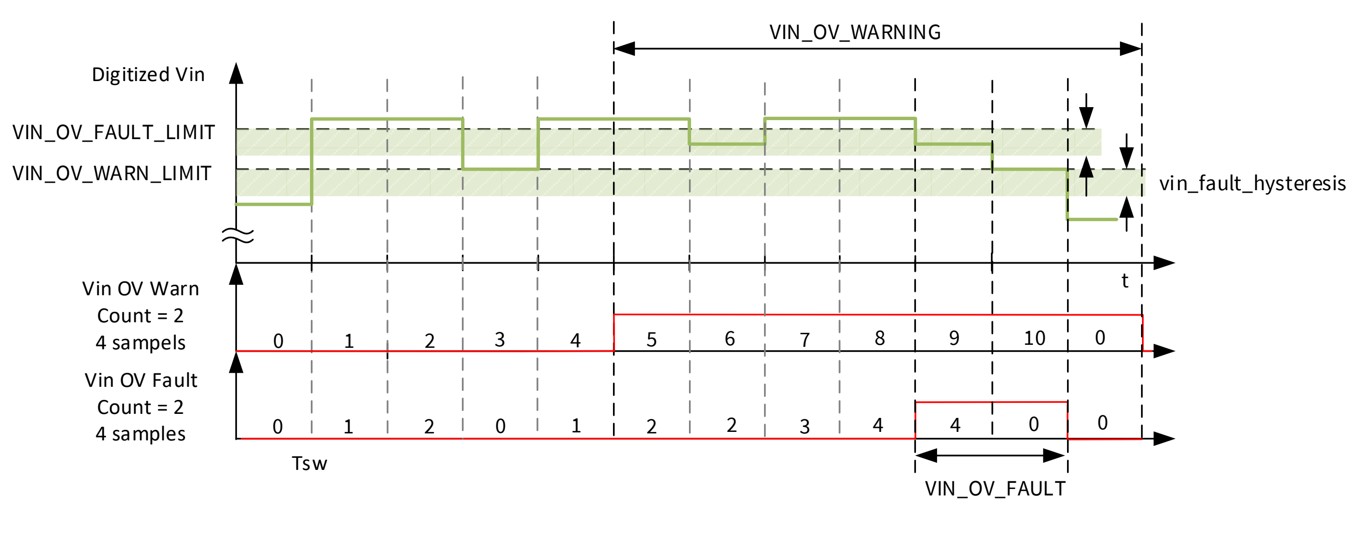


Figure 174 V_{IN} **OV fault hysteresis and fault count**

12.5.3 V_{IN} OVP/UVP fault response

Input overvoltage and undervoltage fault response follows the “voltage, temperature and TON_MAX faults response” as in Table 81. When the input voltage is sensed from the secondary side by V_{RECT} sensing, input overvoltage shutdown will be latched. This is because the V_{RECT} sensing is not able to detect input voltage without switching the primary MOSFETs. Once shut down, it loses the information of V_{IN} and will stay in the off mode. The converter can resume operation by re-enabling after the fault is cleared.

When the input voltage is sensed from the primary side or through an auxiliary power supply, all types of fault response are supported. Please refer to chapter 2 for more information on input voltage sense.

12.5.4 V_{IN} ON/OFF

The VIN_ON and VIN_OFF PMBus commands provide UVLO protection to the system.

Table 96 V_{IN} ON/OFF PMBus command descriptions

Command name	Command code	Description
VIN_ON	35_H	The VIN_ON command sets the value of the input voltage V_{IN}, in volts, at which the device is enabled to start power conversion. Data format = LINEAR 11, U8.2, LSB = 0.25 V, range = 0 to 63.75 V Data format = LINEAR 11, U9.1, LSB = 0.5 V, range = 0 to 127.5 V
VIN_OFF	36_H	The VIN_OFF command sets the value of the input voltage V_{IN}, in volts, at which the unit, once operation has started, should stop power conversion. Data format = LINEAR 11, U8.2, LSB = 0.25 V, range = 0 to 63.75 V Data format = LINEAR 11, U9.1, LSB = 0.5 V, range = 0 to 127.5 V

12.6 Input OCP

12.6.1 I_{IN} OCP PMBus commands

Table 97 describes the PMBus commands relevant to the input OCP.

Table 97 I_{IN} OCP relevant PMBus command descriptions

Command name	Command code	Description
IIN_OC_FAULT_LIMIT	$5B_H$	This command sets the value of the input current, in amps, that causes an input over-current fault. Data format = LINEAR 11, U6.2 LSB 0.25 A, range 0 to 63.75 A
IIN_OC_FAULT_RESPONSE	$5C_H$	This command instructs the device on what action to take in response to an input over-current fault. See section 0 for I_{IN} OC fault response.
IIN_OC_WARN_LIMIT	$5D_H$	This command sets the value of the input current, in amps, that causes an input over-current warning. The IIN_OC_WARN_LIMIT is used as the hysteresis threshold for IIN_OC_FAULT. Data format = LINEAR 11, U6.2 LSB = 0.25 A, range = 0 to 63.75 A

12.6.2 I_{IN} OCP registers

Table 98 I_{IN} OC relevant register descriptions

Name	Address (loop 0/1)	Bits	Description
Fault peripheral			

fault_iin_oc_fault_cnt	7000_3C04_H
	7000_4004_H
		[12:8]	Input over-current fault count. Defines the number of consecutive switching cycles (T_{sw}) by which the low-pass filtered input current must exceed the fault threshold in order to assert a fault.
			Count = fault_iin_oc_fault_cnt +1
			Range = 1 to 32 switching cycles

fault_iin_oc_warn_cnt	7000_3C04_H
	7000_4004_H
		[17:13]	Input over-current warning count. Defines the number of consecutive switching cycles (T_{sw}) by which the low-pass filtered input current must exceed the warning threshold in order to assert a warning.
			Count = fault_iin_oc_fault_cnt +1
			Range = 1 to 32 switching cycles

fault_iin_fault_hyst	7000_3C08_H 7000_4008_H	[9:5]	Input current (I_{IN}) fault (warning) de-assertion hysteresis. Internally, the I_{IN} OV fault is asserted when I_{IN} exceeds IIN_OC_FAULT_LIMIT for fault_iin_oc_fault_cnt samples without dropping below (IIN_OC_FAULT_LIMIT-fault_iin_fault_hyst). The samples above IIN_OC_FAULT_LIMIT need not be consecutive but a single sample below (IIN_OC_FAULT_LIMIT-fault_iin_fault_hyst) will reset the count. For the typical case this parameter should be set to a positive current. This hysteresis parameter applies to all I_{IN} faults and warnings as shown in 0. Resolution = 0.25 A, range = -4 to 3.75 A
Telemetry peripheral			

tlm_kfp_iin	7000_3400_H
	7000_3800_H
		[19:14]	Input current telemetry LPF coefficient index. Note that exponent settings greater than 9 are clamped to 9. Set to 63 to bypass filter.
			kfp_exp = tlm_kfp_iout [5:2]
			kfp_man = 4 + tlm_kfp_iout [1:0]
			kfp = kfp_man * 2^(kfp_exp – 13)
			F3db (kHz) = [kfp/(1-kfp)] * F_{switch} (kHz)/2π
			Range = 0.019 to 30.947 kHz

Table 99 I_{IN} OC hysteresis

Fault	Asserted	De-asserted
IIN_OC_FAULT	> IIN_OC_FAULT_LIMIT	<= (IIN_OC_FAULT_LIMIT-fault_iin_fault_hyst)
IIN_OC_WARN	> IIN_OC_WARN_LIMIT	<= (IIN_OC_WARN_LIMIT-fault_iin_fault_hyst)

Please note the I_{IN} fault is detected after the telemetry filter **tlmX_kfp_iin**. To have fast I_{IN} fault response, bypass the filter by setting it to 63.

12.6.3 I_{IN} OCP fault response

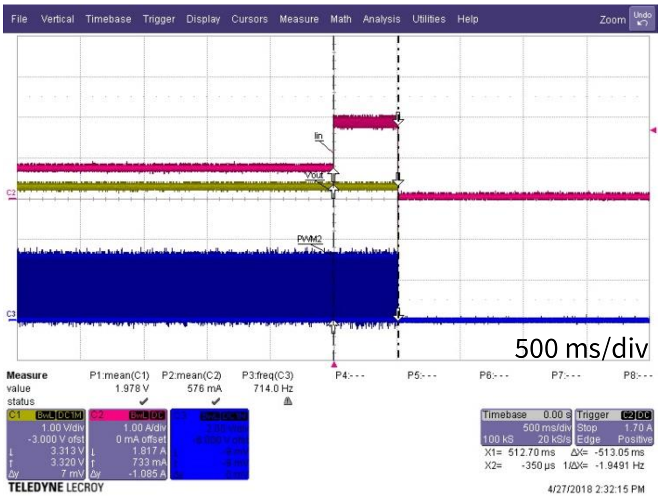
Input over-current fault response can be configured to be one of the following, as in Table 100. The retry setting and delay time can be configured in the GUI by using the drop-down list of the IIN_OC_FAULT_RESPONSE command. Please note that under the same load, the input current changes with input voltage, thus it would have a higher limit at low-line than at high-line. A variable-input over-current limit could be achieved by a FW patch.

Table 100 I_{IN} OC fault response

Response	Description
Ignore	Continues operation without interruption.
Operate for delay time	Continues operation for delay time. If the fault condition is still present at the end of the delay time, the unit responds as programmed in the retry setting.
Disable and retry	Shuts down and responds according to the retry setting.
Disable and resume when OK	Shuts down while the fault is present. The output remains disabled until the fault is cleared.

Figure 175 shows the XDPP1100 waveforms under input over-current fault protection, tested with ACF 3.3 V/30 A board, at 48 V input. The load current was increased from 10 A to 30 A to trigger the input OC fault. The IIN_OC_FAULT_LIMIT was set to 2 A just to demonstrate the IIN OC fault response.

Ch1: V_{OUT} (1 V/div), Ch2: I_{IN} (1 A/div), Ch3: primary PWM (2 V/div)


File Vertical Timebase Trigger Display Cursors Measure Math Analysis Utilities Help Zoom Undo
Iin
Vout
PWM2
C2
C3
500 ms/div
TELEDYNE LECROY

Measure: P1:mean(C1) 1.978 V ✓; P2:mean(C2) 576 mA ✓; P3:freq(C3) 714.0 Hz ⚠; P4:- - -; P5:- - -; P6:- - -; P7:- - -; P8:- - -

C1 BwL DC1M: 1.00 V/div, -3.000 V ofst, ↓ 3.313 V, ↑ 3.320 V, Δy 7 mV

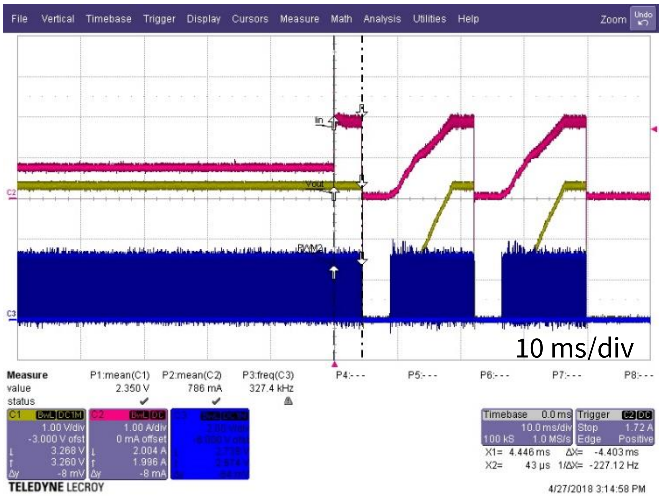
C2 BwL DC: 1.00 A/div, 0 mA offset, ↓ 1.817 A, ↑ 733 mA, Δy -1.085 A

Timebase 0.00 s, 500 ms/div, 100 kS, 20 kS/s; Trigger C2 DC, Stop 1.70 A, Edge Positive

X1= 512.70 ms ΔX= -513.05 ms
X2= -350 µs 1/ΔX= -1.9491 Hz

4/27/2018 2:32:15 PM

Operate for delay time, no retry, 128 unit delay, 4 ms
Delay time = 128*4 ms = 512 ms (measured 513 ms)


File Vertical Timebase Trigger Display Cursors Measure Math Analysis Utilities Help Zoom Undo
Iin
Vout
PWM2
C2
C3
10 ms/div
TELEDYNE LECROY

Measure: P1:mean(C1) 2.350 V ✓; P2:mean(C2) 786 mA ✓; P3:freq(C3) 327.4 kHz ⚠; P4:- - -; P5:- - -; P6:- - -; P7:- - -; P8:- - -

C1 BwL DC1M: 1.00 V/div, -3.000 V ofst, ↓ 3.268 V, ↑ 3.260 V, Δy -8 mV

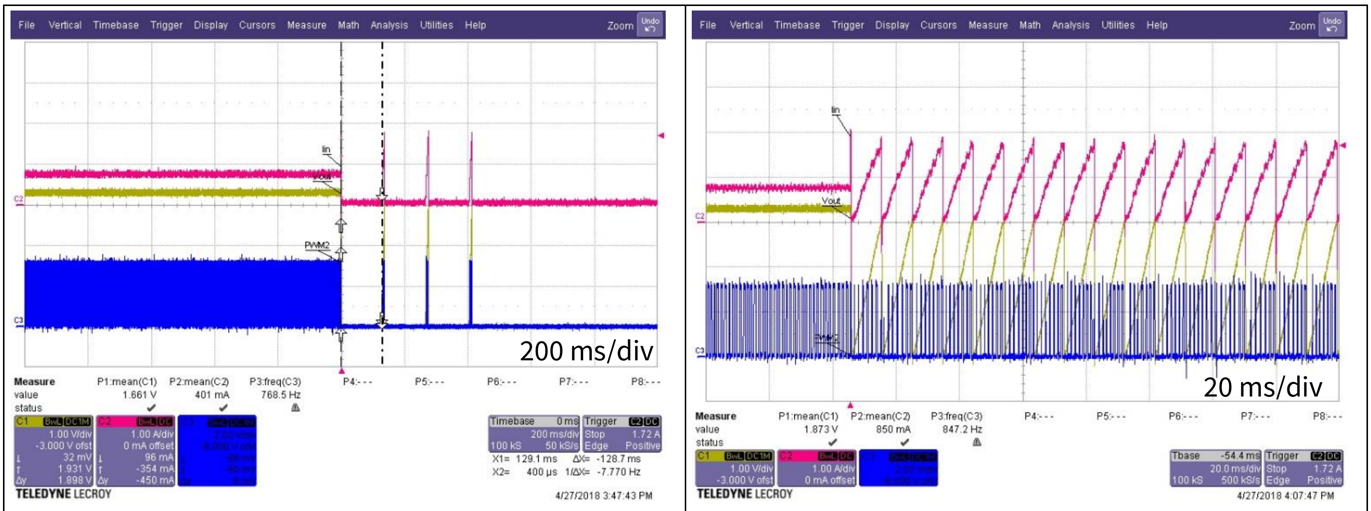
C2 BwL DC: 1.00 A/div, 0 mA offset, ↓ 2.004 A, ↑ 1.996 A, Δy -8 mA

Timebase 0.0 ms, 10.0 ms/div, 100 kS, 1.0 MS/s; Trigger C2 DC, Stop 1.72 A, Edge Positive

X1= 4.446 ms ΔX= -4.403 ms
X2= 43 µs 1/ΔX= -227.12 Hz

4/27/2018 3:14:58 PM

Operate for delay time, retry twice, 4 unit delay, 1 ms
Delay time = 4*1 ms = 4 ms (measured 4.4 ms)



Disable and retry three times, 128 unit delay, 1 ms

Delay time = 128*1 ms = 128 ms (measured 128.7 ms)

Disable and resume when OK

Figure 175 Input over-current fault protection

12.6.4 Input over-power warning

The PMBus command PIN_OP_WARN_LIMIT sets the input power warning threshold. If the input power is higher than the limit, the PIN_OP_WARN will be asserted and bit 0 of the STATUS_INPUT is set to 1.

The maximum input power limit could be set to 4096 W when exponent 2 is selected. If the exponent is 0, the maximum input power limit can be set to 1024 W.

12.7 Overtemperature and undertemperature protection

12.7.1 OT/UT PMBus commands

Table 101 OT and UT relevant PMBus command descriptions

Command name	Command code	Description
OT_FAULT_LIMIT	$4F_H$	This command sets the overtemperature fault threshold, in degrees Celsius. Data format = LINEAR11, U8.0 LSB = 1°C, range = 0 to 255°C
OT_FAULT_RESPONSE	50_H	This command determines the action taken in response to an overtemperature fault.
OT_WARN_LIMIT	51_H	This command sets the threshold, in degrees Celsius, for the overtemperature warning alarm. This value is typically less than the overtemperature fault threshold. It is used as the hysteresis threshold for OT_FAULT_LIMIT. Data format = LINEAR11, U8.0 LSB = 1°C, range = 0 to 255°C
UT_WARN_LIMIT	52_H	This command sets the threshold, in degrees Celsius, for the undertemperature warning alarm. This value is typically higher than the undertemperature fault threshold. It is used as the

Command name	Command code	Description
		hysteresis threshold for UT_FAULT_LIMIT. Data format = LINEAR11, S9.-1 LSB = 2°C, range = -255 to 255°C
UT_FAULT_LIMIT	53_H	This command sets the undertemperature fault threshold, in degrees Celsius. Data format = LINEAR11, S9.-1 LSB = 2°C, range = -256 to 255°C
UT_FAULT_RESPONSE	54_H	This command determines the action taken in response to an undertemperature fault.

12.7.2 OT/UT registers

Table 102 OT and UT relevant register description

Name	Address (loop 0/1)	Bits	Description
Fault peripheral			
fault_temp_fault_hyst	7000_3C08$_H$ 7000_4008$_H$	[26:22]	Temperature fault (warning) de-assertion hysteresis. Internally, the OT fault is asserted when the temperature exceeds the ADC code corresponding to OT_FAULT_LIMIT. The direction of the comparison depends on whether a negative temperature coefficient (NTC) or positive temperature coefficient (PTC) sensor is used. This means the sign of this parameter will also depend on the sensor chosen. Typically, the user should select a negative value for PTC and a positive value for NTC. This hysteresis parameter applies to all temperature faults and warnings as shown in Table 103. LSB = 1 TS ADC code, range = -16 to 15 TS ADC codes
fault_temp_src_sel	7000_3C64$_H$ 7000_4064$_H$	[2:0]	Loop temperature fault source select. 0 ATSEN input NTC 1 BTSEN input NTC 2 Internal temperature (ITSEN) PTC 3 ATSEN input PTC 4 BTSEN input PTC

Table 103 OT/UT hysteresis

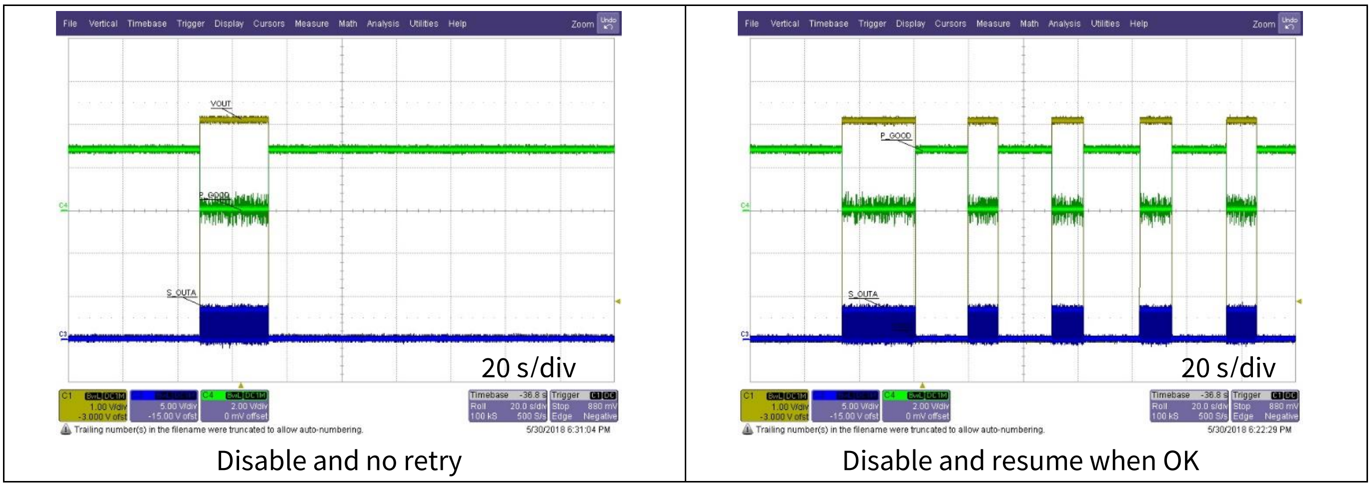
TC	Fault	Asserted	De-asserted
PTC	OT_FAULT	> ADC_Code(OT_FAULT_LIMIT)	< (ADC_Code(OT_FAULT_LIMIT)+fault_temp_fault_hyst)
PTC	OT_WARN	> ADC_Code(OT_WARN_LIMIT)	< (ADC_Code(OT_WARN_LIMIT)+fault_temp_fault_hyst)
PTC	UT_WARN	< ADC_Code(UT_WARN_LIMIT)	> (ADC_Code(UT_WARN_LIMIT)-fault_temp_fault_hyst)
PTC	UT_FAULT	< ADC_Code(UT_FAULT_LIMIT)	> (ADC_Code(UT_FAULT_LIMIT)-fault_temp_fault_hyst)

TC	Fault	Asserted	De-asserted
NTC	OT_FAULT	< ADC_Code(OT_FAULT_LIMIT)	> (ADC_Code(OT_FAULT_LIMIT)+fault_temp_fault_hyst)
NTC	OT_WARN	< ADC_Code(OT_WARN_LIMIT)	> (ADC_Code(OT_WARN_LIMIT)+fault_temp_fault_hyst)
NTC	UT_WARN	> ADC_Code(UT_WARN_LIMIT)	< (ADC_Code(UT_WARN_LIMIT)-fault_temp_fault_hyst)
NTC	UT_FAULT	> ADC_Code(UT_FAULT_LIMIT)	< (ADC_Code(UT_FAULT_LIMIT)-fault_temp_fault_hyst

12.7.3 OTP/UTP fault response

The temperature fault response can be configured as in Table 81. Please note, if the fault response is set to “disable and retry for definite retry number”, the converter will only retry after the temperature crosses the warning limit (FW hysteresis), i.e. when the OT fault is cleared. The temperature usually changes much more slowly than the start-up rise time and won’t reach the fault threshold again during the subsequent restart. Should the converter have a successful start-up, the previous OT fault status will be cleared and the retry number will be reset. Thus, when the OTP or UTP fault response is configured with retry and a retry number, it is usually not able to be tested unless the temperature changes rapidly.

Ch1: V_{OUT} (1 V/div), Ch3: PWM (5 V/div), Ch4: Power good (2 V/div), timescale: 20 s/div



Disable and no retry

Disable and resume when OK

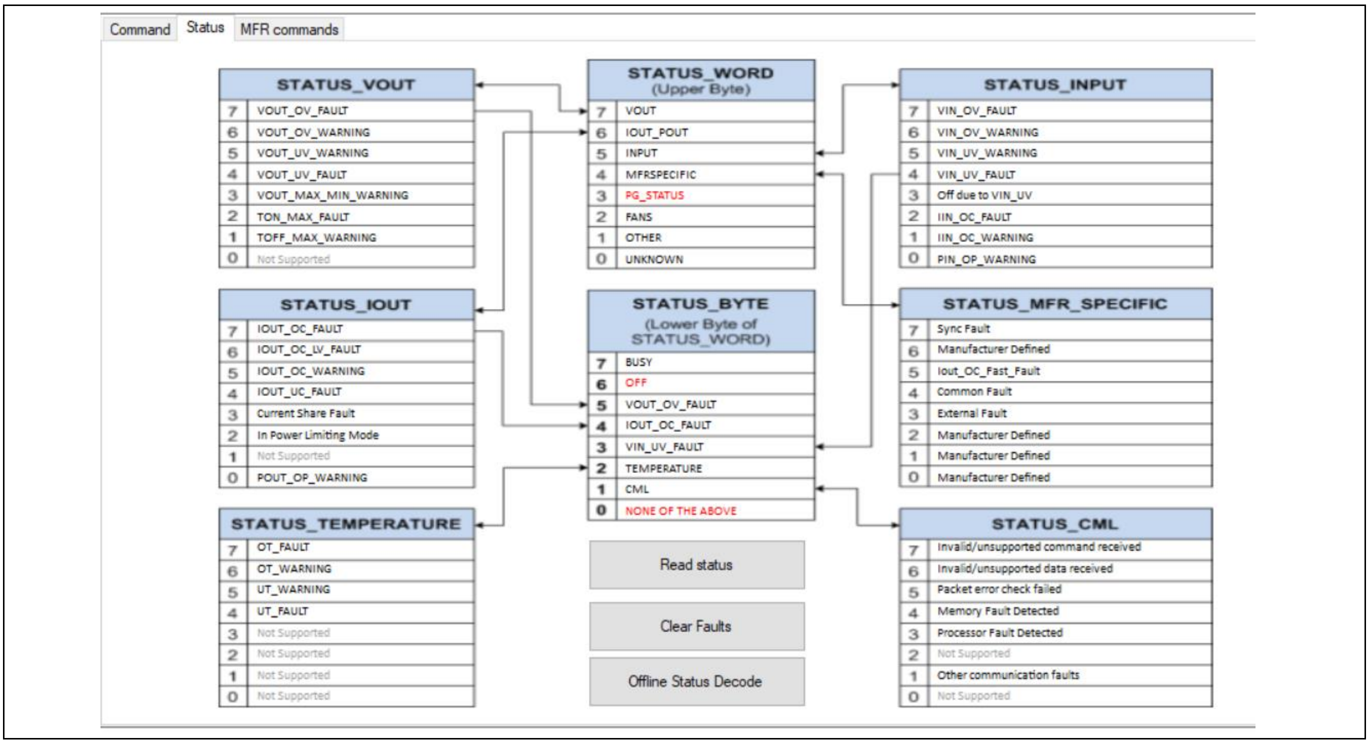
Figure 176 Overtemperature protection waveform (HBCT converter with V_{OUT} = 5 V)

12.8 Fault status reporting and fault mask

12.8.1 Fault status

The XDPP1100 GUI has a status reporting page that lists the status of input/output voltage, current and temperature conditions. If any fault or warning is triggered, the corresponding bit of the status will turn red, indicating the fault. To refresh the status, clicking the “Read status” button. If the fault is no longer present, clicking the “Clear Faults” button will write the CLEAR_FAULTS command and reset the color of the status bit. If the fault persists, clicking the “Clear Faults” button will write the CLEAR_FAULTS command but the fault will trip again and the color of the status bit remains red.

Please note that the CLEAR_FAULTS command can’t be written if the write protect mode is enabled by command 0x10 WRITE_PROTECT.



STATUS_VOUT	
7	VOUT_OV_FAULT
6	VOUT_OV_WARNING
5	VOUT_UV_WARNING
4	VOUT_UV_FAULT
3	VOUT_MAX_MIN_WARNING
2	TON_MAX_FAULT
1	TOFF_MAX_WARNING
0	Not Supported

STATUS_IOUT	
7	IOUT_OC_FAULT
6	IOUT_OC_LV_FAULT
5	IOUT_OC_WARNING
4	IOUT_UC_FAULT
3	Current Share Fault
2	In Power Limiting Mode
1	Not Supported
0	POUT_OP_WARNING

STATUS_TEMPERATURE	
7	OT_FAULT
6	OT_WARNING
5	UT_WARNING
4	UT_FAULT
3	Not Supported
2	Not Supported
1	Not Supported
0	Not Supported

STATUS_WORD (Upper Byte)	
7	VOUT
6	IOUT_POUT
5	INPUT
4	MFRSPECIFIC
3	PG_STATUS
2	FANS
1	OTHER
0	UNKNOWN

STATUS_BYTE (Lower Byte of STATUS_WORD)	
7	BUSY
6	OFF
5	VOUT_OV_FAULT
4	IOUT_OC_FAULT
3	VIN_UV_FAULT
2	TEMPERATURE
1	CML
0	NONE OF THE ABOVE

STATUS_INPUT	
7	VIN_OV_FAULT
6	VIN_OV_WARNING
5	VIN_UV_WARNING
4	VIN_UV_FAULT
3	Off due to VIN_UV
2	IIN_OC_FAULT
1	IIN_OC_WARNING
0	PIN_OP_WARNING

STATUS_MFR_SPECIFIC	
7	Sync Fault
6	Manufacturer Defined
5	Iout_OC_Fast_Fault
4	Common Fault
3	External Fault
2	Manufacturer Defined
1	Manufacturer Defined
0	Manufacturer Defined

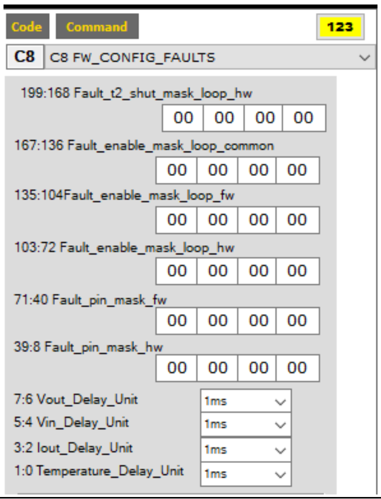
STATUS_CML	
7	Invalid/unsupported command received
6	Invalid/unsupported data received
5	Packet error check failed
4	Memory Fault Detected
3	Processor Fault Detected
2	Not Supported
1	Other communication faults
0	Not Supported

Figure 177 GUI status report

12.8.2 Fault mask

PMBus command 0XC8 FW_CONFIG_FAULTS is a 25-byte command that allows the user to set a fault mask according to the following description.

- Bits [199:168], Fault_t2_shut_mask_loop_hw, “Masking for loop hw faults shutdown on t2”
- Bits [167:136], Fault_enable_mask_loop_common, “Masking for loop common faults enable”
- Bits [135:104], Fault_enable_mask_loop_fw, “Masking for loop fw faults enable”
- Bits [103:72], Fault_enable_mask_loop_hw, “Masking for loop hw faults enable”
- Bits [71:40], Fault_pin_mask_fw, “Masking for firmware faults”
- Bits [39:8], Fault_pin_mask_hw, “Masking for hardware faults”



C8 C8 FW_CONFIG_FAULTS

199:168 Fault_t2_shut_mask_loop_hw
00 00 00 00

167:136 Fault_enable_mask_loop_common
00 00 00 00

135:104Fault_enable_mask_loop_fw
00 00 00 00

103:72 Fault_enable_mask_loop_hw
00 00 00 00

71:40 Fault_pin_mask_fw
00 00 00 00

39:8 Fault_pin_mask_hw
00 00 00 00

7:6 Vout_Delay_Unit 1ms
5:4 Vin_Delay_Unit 1ms
3:2 Iout_Delay_Unit 1ms
1:0 Temperature_Delay_Unit 1ms

Figure 178 FW_CONFIG_FAULTS

The HW fault mask (fault_t2_shut_mask_loop_hw, fault_enable_mask_loop_hw and fault_pin_mask_hw) has the bit assignment shown in Table 104.

An individual fault is disabled when the corresponding bit of the fault_enable_mask_loop_hw is high. It disables both the fault response and the PMBus fault reporting.

The fault_pin_mask_hw is used to disable the individual HW fault from triggering the fault pin when the corresponding bit of fault_pin_mask_hw is high.

Table 104 HW fault mask bit assignment

Bit	Fault	Bit	Fault
0	Reserved	1	VOUT_OV_FAULT
2	VOUT_OV_WARN	3	VOUT_UV_FAULT
4	VOUT_UV_WARN	5	VIN_OV_FAULT
6	VIN_OV_WARN	7	VIN_UV_FAULT
8	VIN_UV_WARN	9	IOUT_OC_FAULT
10	IOUT_OC_LV_FAULT	11	IOUT_OC_WARN
12	IOUT_UC_FAULT	13	MFR_IOUT_OC_FAST
14	IIN_OC_FAULT	15	IIN_OC_WARN
16	OT_FAULT	17	OT_WARN
18	UT_FAULT	19	UT_WARN
20	POWER_LIMIT_MODE	21	ISHARE_FAULT
22	VOUT_MAX_MIN_WARN	23	SYNC_FAULT
24-31	Unused		

For example, to disable the fault pin responding to UT_WARN, set bit 19 to 1 by writing 0x 00 08 00 00 to fault_pin_mask_hw. The fault pin will not response on UT warning, but the status still reports the UT_WARN fault when it is triggered. The user could read STATUS_TEMPERATURE and see the UT_WARNING. To disable UT WARN reporting, write 0x 00 08 00 00 to fault_enable_mask_loop_hw.

The FW fault mask (fault_enable_mask_loop_fw and fault_pin_mask_fw) has the bit assignment shown in Table 105. An individual fault is disabled when the corresponding bit of the fault_enable_mask_loop_fw is high. It disables both the fault response and the PMBus fault reporting. The fault_pin_mask_fw is used to disable the individual FW fault to trigger the Fault pin when the corresponding bit of the fault_pin_mask_fw is high.

Table 105 FW fault mask bit assignment

Bit	Fault	Bit	Fault
0	RESERVED_FW_NO_FAULT	1	COMMON_SHUTDOWN
2	TON_MAX_FAULT	3	TOFF_MAX_WARN
4	PIN_OP_WARN	5	POUT_OP_WARN
6	VIN_INSUFFICIENT	7 to 31	Spares for user

For example, to disable the POUT_OP_WARN, set bit 5 of fault_enable_mask_loop_fw to 1 by writing 0x 00 00 00 20. If the output power is higher than the POUT_OP_WARN_LIMIT, the output over-power warning won't be reported in STATUS_IOUT.

The common fault mask bit assignment is listed in Table 106. The individual common fault will be disabled if the corresponding bit of fault_enable_mask_loop_common is high. The detailed description of common faults can be found in section 12.9.2.

Table 106 Common fault mask bit assignment

Bit	Fault	Bit	Fault
0	Unused	1	Unused
2	IS1 (ISEN) tracking fault	3	IS2 (BISEN) tracking fault
4	Fbal1 fault	5	IS1 (ISEN) PCL fault
6	IS1 (ISEN) SCP fault	7	Fbal2 fault
8	IS2 (BISEN) PCL fault	9	IS2 (BISEN) SCP fault
10	Unused	11	VREF open fault
12	VSEN open fault	13	Unused
14	VRREF open fault	15	VRSEN open fault
16	Unused	17	BVREF_BVRREF open fault
18	BVSEN_BVRSEN open fault	19 to 31	Unused

12.8.2.1 Masking for loop HW faults shutdown on T2

T2 shutdown is a feature which means that, at converter shutdown, all PWMs including primary control PWMs and secondary SR PWMs are turned off at the falling edge of the primary PWM. **Figure 179** shows the primary PWM (Q1, Q2), SR and output inductor current in T2 shutdown. T2 shutdown prevents negative current flow in the SR MOSFETs when the primary MOSFETs are turned off, thus reducing voltage spikes on the SR MOSFETs.

The shutdown could be a user-initiated system turn-off, or it could be a fault shutdown on fault detection. Setting the individual bit of the fault_t2_shut_mask_loop_hw register high enables T2 shutdown of the corresponding fault. To enable T2 shutdown of all the HW faults, write FF FF FF FF to fault_t2_shut_mask_loop_hw.

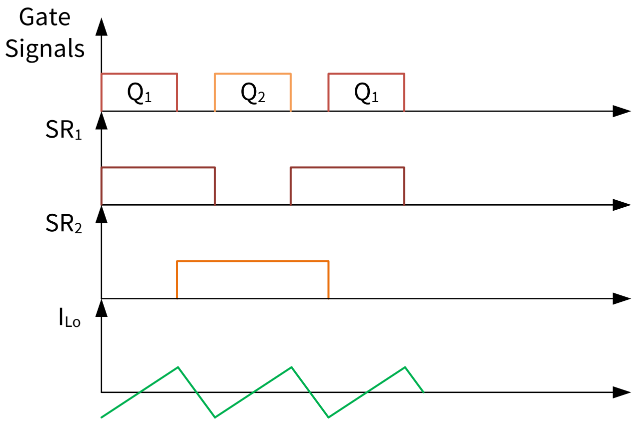
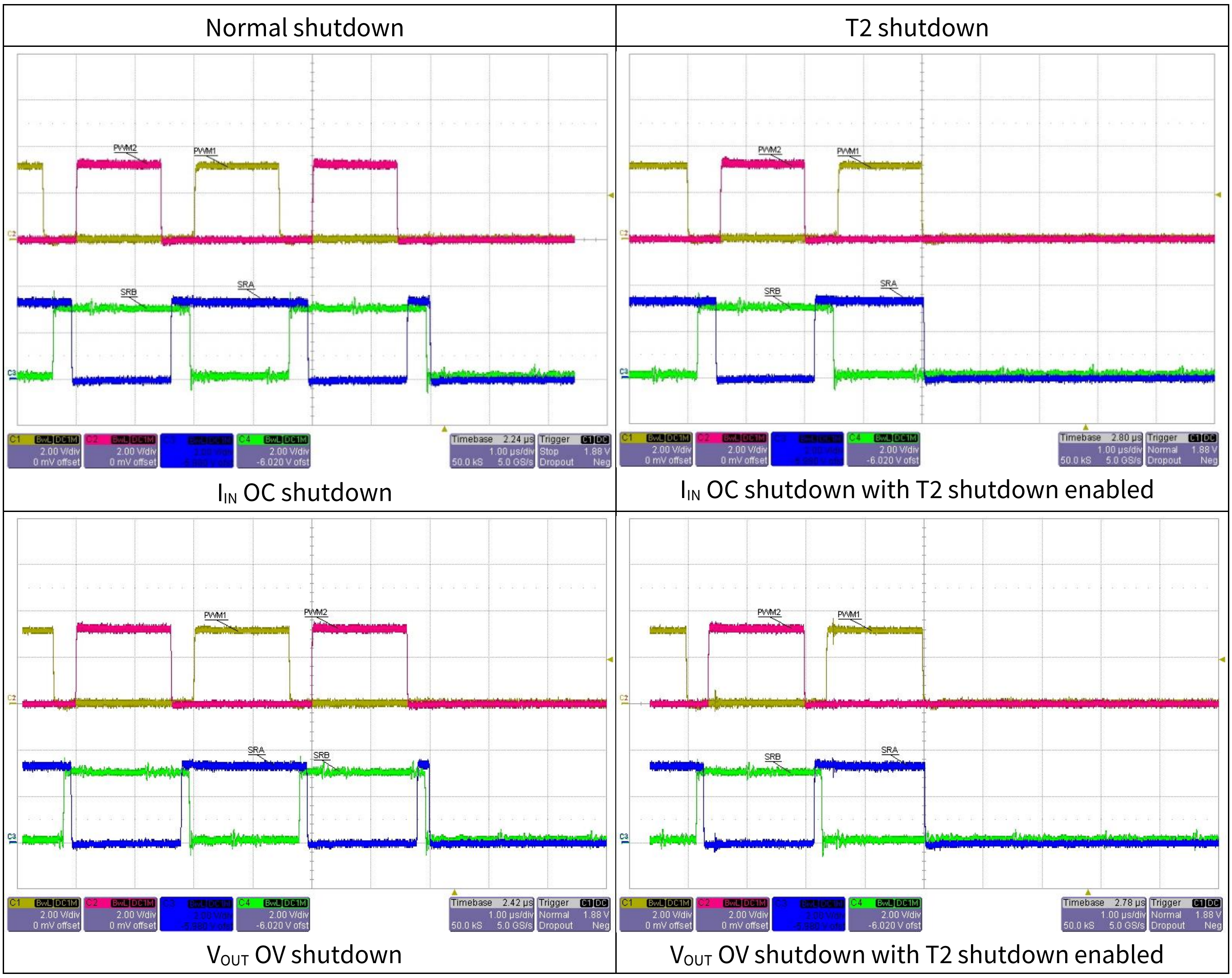


Figure 179 SR shutdown sequence example waveforms (HBCT topology)

Figure 180 shows the difference between a normal shutdown and a T2 shutdown of a FBFW converter.

Ch1: primary PWM1, Ch2: primary PWM2, Ch3: SR PWM SRA, Ch4: SR PWM SRB



I_{IN} OC shutdown

I_{IN} OC shutdown with T2 shutdown enabled

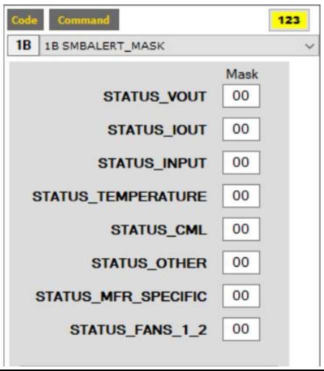
V_{OUT} OV shutdown

V_{OUT} OV shutdown with T2 shutdown enabled

Figure 180 T2 shutdown waveforms

12.8.3 SMBALERT_MASK

PMBus command 0x1B SMBALERT_MASK is used to mask warning or fault conditions from asserting the SMBALERT signal. The configuration of the SMBALERT_MASK in GUI is shown in **Figure 181**.



1B | 1B SMBALERT_MASK

	Mask
STATUS_VOUT	00
STATUS_IOUT	00
STATUS_INPUT	00
STATUS_TEMPERATURE	00
STATUS_CML	00
STATUS_OTHER	00
STATUS_MFR_SPECIFIC	00
STATUS_FANS_1_2	00

Figure 181 SMBALERT_MASK configuration

The SMBALERT is an active low signal. The output should be high if no warning or fault is asserted and it will be pulled down if a warning or fault is asserted. To mask a warning or fault condition, write the corresponding bit to 1.

12.8.4 Simultaneous fault response

When simultaneous faults been triggered, the shutdown response has higher priority. For example, when both OT_FAULT and VOUT_UV_FAULT are triggered, the OT_FAULT response is “disable and resume when OK”, while the VOUT_UV_FAULT response is “ignore”, so the controller follows the OT_FAULT response “disable and resume when OK”.

If both faults have a shutdown response the priority goes by FAULT_TYPE enum list. The top fault has higher priority. For example, VOUT_OV_FAULT has higher priority than IOUT_OC_FAULT and IOUT_OC_FAULT has higher priority than OT_FAULT. If more than one shutdown fault is triggered, the fault response that has more retry numbers has higher priority.

FAULT_TYPE enum list, priority from high to low:

FAULT_TYPE_VOUT_OV_FAULT

FAULT_TYPE_VOUT_OV_WARN

FAULT_TYPE_VOUT_UV_FAULT

FAULT_TYPE_VOUT_UV_WARN

FAULT_TYPE_VIN_OV_FAULT

FAULT_TYPE_VIN_OV_WARN

FAULT_TYPE_VIN_UV_FAULT

FAULT_TYPE_VIN_UV_WARN

FAULT_TYPE_IOUT_OC_FAULT

FAULT_TYPE_IOUT_OC_LV_FAULT

FAULT_TYPE_IOUT_OC_WARN

FAULT_TYPE_IOUT_UC_FAULT

FAULT_TYPE_MFR_IOUT_OC_FAST

FAULT_TYPE_IIN_OC_FAULT

FAULT_TYPE_IIN_OC_WARN

FAULT_TYPE_OT_FAULT

FAULT_TYPE_OT_WARN

FAULT_TYPE_UT_FAULT

FAULT_TYPE_UT_WARN

FAULT_TYPE_IN_POWER_LIMITING_MODE

FAULT_TYPE_CURRENT_SHARE_FAULT

FAULT_TYPE_VOUT_MAX_MIN_WARN

FAULT_TYPE_SYNC_FAULT

It is suggested to avoid simultaneous faults being reported. To avoid simultaneous faults, the register **fault_block_on_shut** must be set to 1. It will block new faults after initial shutdown until FW has completed shutdown-related clean-up. Enabling the **fault_block_on_shut** blocks other faults for about 10 µs to 20 µs. If there are new faults after the blocking time, they can be processed.

12.9 MFR specific fault protection

The XDPP1100 provides the following MFR specific fault protections and status reporting.

- Sync fault
- I_{OUT} OC fast fault
- Common fault

The IOUT_OC_FAST_FAULT is described in chapter 12.4.2. This chapter will introduce the sync fault and common fault.

12.9.1 Sync fault

The sync-in feature enables synchronizing the PWM of the XDPP1100 to an external signal. The sync-in signal should be a square waveform with 50 percent duty-cycle. Any GPIO pins of the XDPP1100 can be mapped to perform the sync function by setting the function register. The introduction of the sync-in feature can be found in chapter 10.1. When the XDPP1100 is synchronized to an external signal, the maximum initial lock range of the sync-in signal is ±6.25 percent of the switching frequency set by the PMBus command FREQUENCY_SWITCH. Once the sync is locked, the system can remain in sync with the input signal up to a maximum variation of ±12.5 percent of the FREQUENCY_SWITCH.

If the sync-in signal is out of the maximum sync range, and the XDPP1100 has failed to lock to the incoming signal, the sync fault will be reported in the STATUS_MFR_SPECIFIC bit [7].

12.9.2 Common fault

The XDPP1100 supports the following common faults:

- IS (ISEN, BISEN), tracking fault
- Fbal fault, (flux balancing fault)
- IS (ISEN, BISEN), PCL fault (peak current limit)
- IS (ISEN, BISEN), SCP fault (short circuit protection)
- VSEN, VREF, VRSEN, VRREF, BVSEN_BVRSEN, BVREF_BVRREF, open fault

Please refer to Table 106 for the bit assignment of the common faults. The details of the flux balancing fault are described in chapter 6.1.7, and the IS PCL fault (cycle-by-cycle PCL) is described in chapter 12.4.7, so will not be repeated here.

The response to a common fault can be set to either ignore or shut down. This is configured by the common fault shutdown mask register **fault_shut_mask_com**. The GUI design tool “Fault & Protections” provides an easy way to configure each common fault. As shown in **Figure 162**, checking the box of a fault setting of the corresponding bit to 1 enables fault shutdown. For example, to enable the loop 0 flux balance fault shutdown, check the box of “Fbal1 fault”.

If the **fault_shut_mask_com** is 00000000, the XDPP1100 will not respond to any common fault. The common fault will still be reported through the STATUS_MFR_SPECIFIC once triggered. The user can debug the system by using PMBus command MFR_FIRMWARE_COMMAND and the MFR_FIRMWARE_COMMAND_DATA. An example can be found in chapter 12.9.2.3.

12.9.2.1 Current ADC tracking fault

The current sense of the XDPP1100 is achieved by the combination of direct current sensing and current emulation. The XDPP1100 predicts the rising and falling slope of the current signal to pre-position the ADC output. The current rising and falling slope is estimated based on the PWM on/off time, the output inductance, the input magnetizing inductance, and the instantaneous input and output voltage. Refer to chapter 3 for the details of current sense. If the measured current signal doesn't match with the emulated current, the current tracking fault can be reported. The fault threshold is defined by error ratio.

Table 107 SCP register description

Name	Address (loop 0/1)	Bits	Description
ISEN peripheral			
isp_track_fault_en	7000_2414_H 7000_2814_H	[0]	Current sense tracking fault enable. The tracking fault detects the inability of the current sense emulator to track the incoming current sense signal. This is typically an indication of a board problem (e.g., missing sense resistor, bad pin connection). = 0, disabled = 1, enabled
isp_err_ratio_sel	7000_2414_H 7000_2814_H	[3:1]	Current sense tracking fault error ratio select. = 0, 11.1 percent threshold = 1, 20 percent threshold = 2, 27.3 percent threshold = 3, 33.3 percent threshold = 4, 42.9 percent threshold = 5, 50 percent threshold = 6, 60 percent threshold = 7, 66.7 percent threshold

12.9.2.2 SCP

The IS_SCP is used for short circuit protection. The description of the SCP register is shown in Table 108. SCP trips when the phase current (i.e. including ripple) goes above the SCP threshold for a single ADC sample AND the primary-side PWM is off. This means the IADC tracking feedback is causing the over-current, and not the emulation.

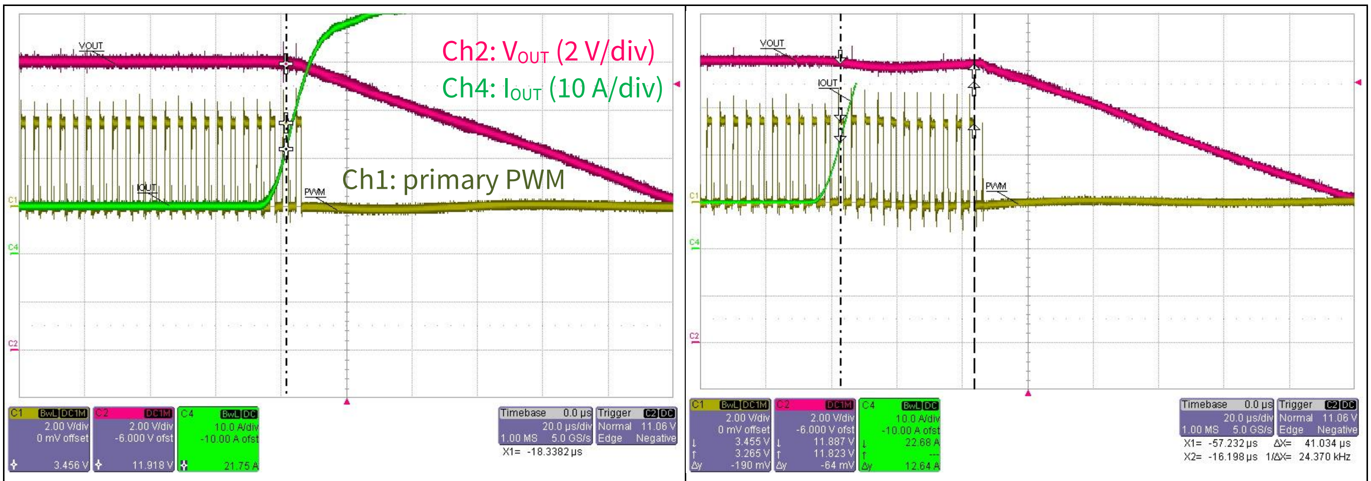
Table 108 SCP register description

Name	Address (loop 0/1)	Bits	Description
ISEN peripheral			
isp_scp_thresh	7000_2400_H 7000_2800_H	[7:0]	SCP fault threshold. The SCP threshold should be set highest among the various current protection

Name	Address (loop 0/1)	Bits	Description
			thresholds as it requires only a single sample above the threshold to trip. The SCP threshold is applied per phase on multiphase topologies. Set this threshold to 0 to disable the fault detection. LSB = 1 A, range = 0 to 255 A

The SCP fault provides the fastest over-current fault protection at 25 MHz sample rate. The SCP response can be configured to ignore or shut down. This is configured by register **fault_shut_mask_com** (common fault shutdown mask), and setting the corresponding bit to 1 enables the fault shutdown. To enable the ISEN short-circuit (SCP) shutdown, write bit [6] of the **fault_shut_mask_com** to 1, i.e. write 00000040$_H$. When the short-circuit happens and triggers the SCP fault, the STATUS_MFR_SPECIFIC will report “common fault”. The user can debug the fault by using PMBus command MFR_FIRMWARE_COMMAND and the MFR_FIRMWARE_COMMAND_DATA. Write 23$_D$ to PMBus command 0xFE MFR_FIRMWARE_COMMAND and read 0xFD MFR_FIRMWARE_COMMAND_DATA. It should return = 00 00 00 40$_H$ on ISEN SCP fault.

Figure 182 shows the different shutdown speed when the converter is shutdown by the SCP or by the fast OCP. The SCP fault threshold is set to 22 A. The MFR_IOUT_OC_FAST_FAULT_LIMIT is set to 20 A. Both fault responses set to shutdown. The **fault0_mfr_iout_oc_fast_cnt** is set to 8 to have 9 counts. The OC_FAST fault block has one switching cycle latency, thus the shutdown happens after 10 switching cycles.



Shutdown by SCP, responded after 1 switching cycle

Shutdown by MFR_IOUT_OC_FAST, responded after 10 switching cycles

Figure 182 Output short circuit protection (FB-FB converter at 48 V input, 12 V output)

12.9.2.3 Voltage open-sense fault detection

In a power supply, if the feedback loop is broken (i.e. the sense resistor is open or short), the controller will not be able to regulate output voltage to the target level. Since the controller doesn’t know the output condition it could be continuously switching with the maximum allowed duty-cycle, which could result in destructive failure due to overvoltage.

A HBCT topology is shown in **Figure 183** as an example. The potential failures are listed in Table 109. The VRSEN pin and R3, R4 are used to sense the input voltage through V_{RECT} sensing. For details of input voltage sense and feed-forward please refer to chapter 2.

Table 109 Feedback open/short failure

Device	Condition	Result
R1	Open	Controller sense V_{OUT} = 0 V, actual **V_{OUT} overvoltage**
R1	Short	Controller sense V_{OUT} without proper scaling, actual V_{OUT} undervoltage
R2	Open	Controller sense V_{OUT} without proper scaling, actual V_{OUT} undervoltage
R2	Short	Controller sense V_{OUT} = 0 V, actual **V_{OUT} overvoltage**
R3	Open	Controller sense V_{RECT} = 0 V, system may shut down due to VIN_OFF
R3	Short	VRSEN pin = V_{IN}/n, potential **damage IC** due to overstress
R4	Open	VRSEN pin = V_{IN}/n, potential **damage IC** due to overstress
R4	Short	Controller sense V_{RECT} = 0 V, system may shutdown due to VIN_OFF

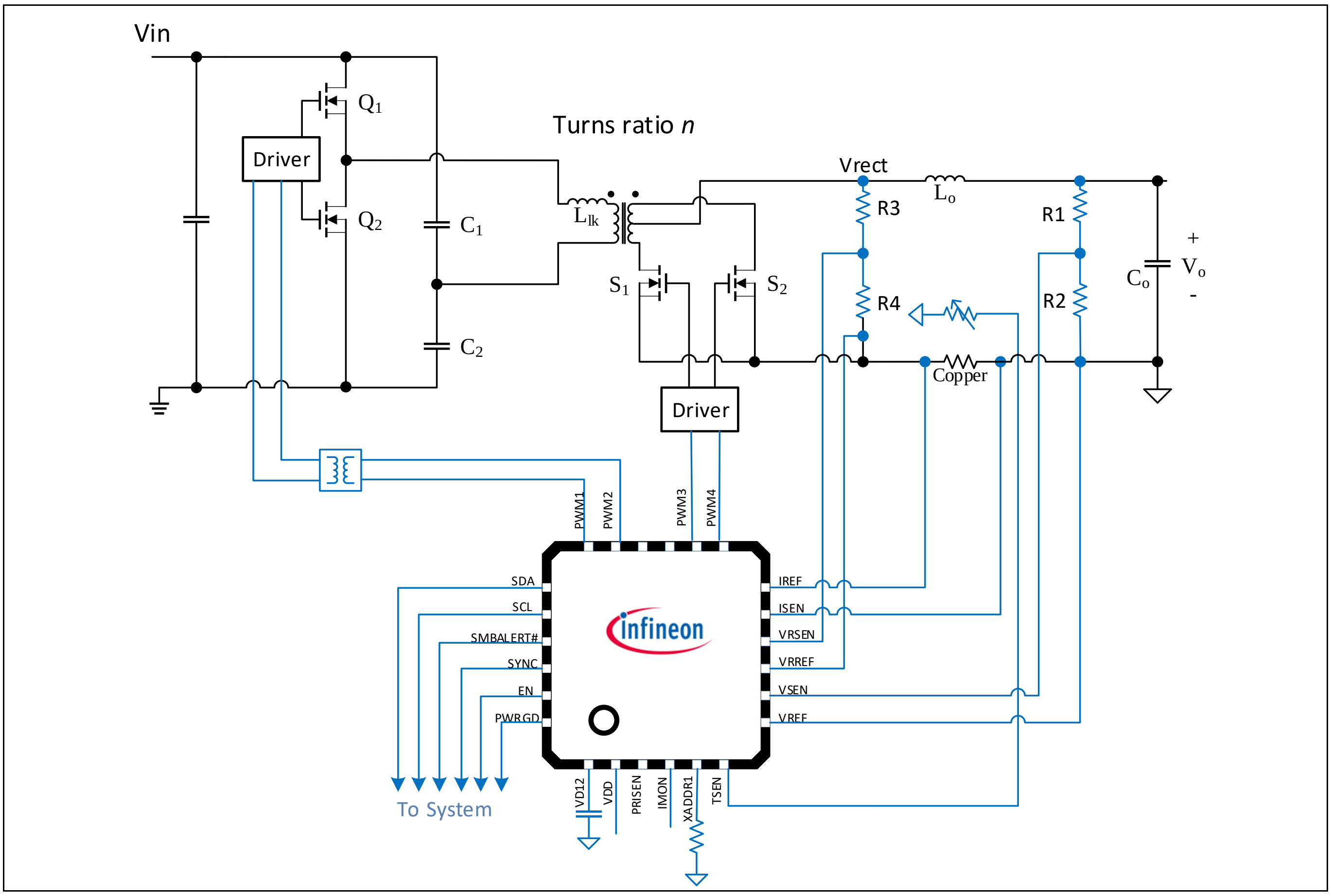


Figure 183 HBCT topology

Some designs use a redundant voltage sensing circuit to separate the output feedback loop and the output overvoltage protection. This approach requires additional components, including voltage divider resistors and comparators. To avoid adding to system cost while still providing reliable operation, the XDPP1100 offers voltage open sense protection (OSP). The protection is implemented in two ways that cover the open fault protection under two different conditions: before converter start-up and during start-up ramp.

To detect the open sense fault before converter start-up, the XDPP1100 enables a 50 µA (+/-15 percent) current source at the input pins of the VADC then measures the voltage. If the voltage is higher than the set threshold indicating a higher than expected impedance, the open fault is asserted. The user could decide to disable the operation when an open fault is detected.

Table 110 OSP by current source register description

Name	Address (loop 0/1)	Bits	Description
VSEN peripheral			
vsp_osp_thresh	7000_080C$_H$ 7000_0C0C$_H$ 7000_100C$_H$	[11:0]	OSP fault threshold. Set to 0 to disable. LSB = 1.25 mV, range = 0 to 5.11875 V

Setting the **vsp_osp_thresh** to a non-zero value will enable the OSP current source during the TON_DELAY time when the converter is enabled for the first time. If the OSP checking of all three VADCs is enabled, the XDPP1100 sources the 50 µA current in the sequence shown in **Figure 184**. Please note that the current source-based OSP detection is only performed at start-up the first time.

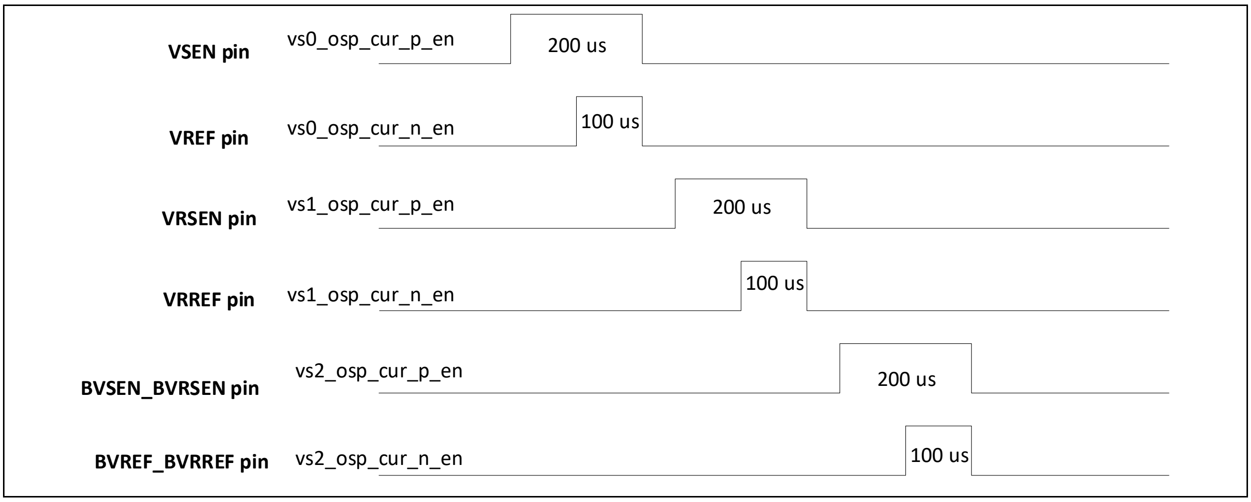


Figure 184 OSP current source sequencing

For example, for a converter with 12 V output voltage, the feedback resistor R1 = 10 kΩ, R2 = 1.1 kΩ. The feedback is connected to the VSEN pin. The V_{RECT} resistor divider has R3 = 12.7 kΩ, R4 = 1 kΩ. The V_{RECT} resistor divider is connected to the VRSEN pin. Both OSPs are enabled.

The XDPP1100 will source 50 µA current out of the VSEN pin first. The current source will turn-on for 200 µs. With both R1 and R2 resistors in place, R1 is in series with R3, R4 then connected to ground. The voltage at the VSEN pin is expected to be: 50 µA * [(10 k+12.7k+1k)/1.1k] = 50 µA * 1.05 kΩ = 52.5 mV.

The voltage of VSEN is shown in Table 111 at all the possible R1, R2 fault conditions. According to this table, the vsp0_osp_thresh (for VSEN) can be set to 100 mV to detect the R2 open fault condition.

$$\text{vsp0\_osp\_thresh} = \frac{100mV}{1.25mV} = 80 \qquad (11.2)$$

The XDPP1100 then sources 50 µA current out of the VREF pin for 100 µs. If the VREF pin is correctly grounded, the VADC would measure the differential voltage between VSEN and VREF. If the VREF pin is floating, the 50 µA current would charge the VREF pin up closer to V_{DD} supply voltage. When the VREF voltage is higher than VSEN, the ADC output clamps to 0 and the XDPP1100 reports VREF OSP fault.

Once the VSEN/VREF OSP detection is done, the XDPP1100 turns off the VSEN and VREF current source and turns on the 50 µA current of the VRSEN pin. The normal voltage at VRSEN is expected to be 48 mV and the abnormal voltages are listed in Table 112. The **vsp1_osp_thresh** is set to 80 for 100 mV threshold.

Table 111 VSEN voltage under open/short failure (R1 = 10 kΩ, R2 = 1.1 kΩ, R3 = 12.7 kΩ, R4 = 1 kΩ)

Device	Condition	VSEN voltage
	Normal	52.5 mV, R2 is in parallel with (R1+R3+R4)
R1	Open	55 mV, only the 1.1 kΩ R2 is connected to the VSEN pin
R1	Short	50.9 mV, R2 is in parallel with (R3+R4)
R2	Open	1185 mV, R1 is in series with (R3+R4)
R2	Short	0 V

Table 112 VRSEN voltage under open/short failure (R1=10 kΩ, R2 = 1.1 kΩ, R3=12.7 kΩ, R4 = 1 kΩ)

Device	Condition	VRSEN voltage
	Normal	48 mV, R4 is in parallel with (R3+R1+R2)
R3	Open	50 mV, only R4 is connected to the VSEN pin
R3	Short	45.8 mV, R4 is in parallel with (R1+R2)
R4	Open	1190 mV, R3 is in series with (R1+R2)
R4	Short	0 V

For the XDPP1100-Q040 that offers three VADCs, the OSP of the third ADC BVSEN_BVRSEN/BVREF_BVRREF can be checked as well. When any of the ADC sense pins detect an OSP fault, the common fault will be reported at PMBus bit [4] of STATUS_MFR_SPECIFIC, and the XDPP1100 won't enable PWM output. The OSP common fault status can be checked by writing 25_D to PMBus command 0xFE MFR_FIRMWARE_COMMAND, or selecting "READ_OPEN_SENSE_COMMON_FAULTS_STATUS" from the drop-down list of the command. Then read the result from PMBus command 0xFD MFR_FIRMWARE_COMMAND_DATA. Upon a common fault detection, the corresponding bit will be set to 1 as in Table 113.

Table 113 OSP common fault bit assignment

Bit	Fault	Bit	Fault
0	VSEN	1	VREF
2	VRSEN	3	VRREF
4	BVSEN_BVRSEN	5	BVREF_BVRREF

It is clear that the resistor short situations cannot be detected by the current source method. The high-side open is also hard to detect if the high-side resistor is much larger than the low-side resistor, with the voltage in the high-side open situation very close to normal. In these cases, the failure could be checked by "PID duty"-based OSP fault detection.

The "PID duty"-based OSP registers are listed in Table 114. The pid0 reports OSP fault on the VSEN open, and the pid1 reports OSP fault on the BVSEN open. The **pid_osp_ff_thr** and the **pid_osp_duty_thr** set the minimum duty-cycle above which to begin checking the OSP fault. The **pid_osp_ff_scale** sets the fault threshold.

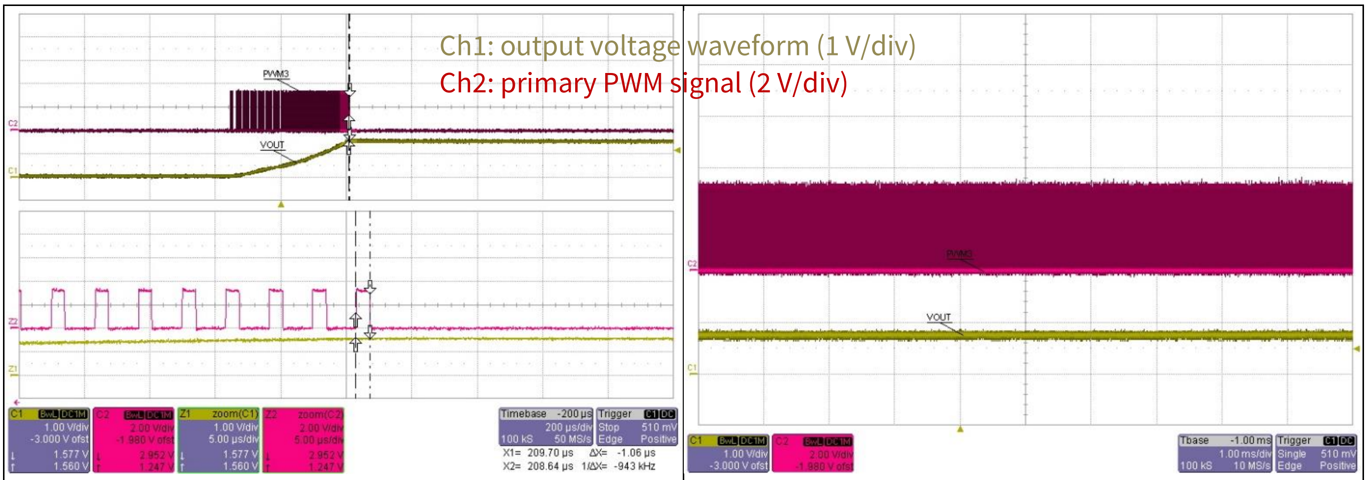
Table 114 OSP by PID register description

Name	Address (loop 0/1)	Bits	Description
pid_osp_ff_thr	7000_1C18_H 7000_2018_H	[6:0]	Defines the minimum PID feed forward value, above which to begin checking for an open sense fault during soft-start.

Name	Address (loop 0/1)	Bits	Description
			LSB = 2^-7, range = 0 to 0.9922
pid_osp_duty_thr	7000_1C18_H 7000_2018_H	[13:7]	Defines the minimum PID duty-cycle value, above which to begin checking for an open sense fault during soft-start. LSB = 2^-7, range = 0 to 0.9922
pid_osp_ff_scale	7000_1C18_H 7000_2018_H	[17:14]	Scale factor applied to PID feed forward term to detect an open sense fault between V_{OUT} and V_{OUT} sense during soft-start. A setting of 0 will disable this fault check. LSB = 0.5, range = 0 to 7.5

For example, the 3.3 V/30 A ACF converter is tested at 36 V input. V_{RECT} voltage is 6 V with 6:1 transformer turns ratio. Switching frequency is 300 kHz. Set **pid_osp_ff_scale** = 3 to get the fault scale factor to 1.5, **pid_osp_duty_thr** = 38 for the OSP starts checking at 29.64 percent duty-cycle, **pid_osp_ff_thr** = 1 sets the feed-forward duty-cycle threshold to 0.78 percent. TON_RISE = 10 ms in this test.

Figure 185 (a) is the start-up waveform with R2 shorted. During the start-up ramp, the XDPP1100 starts looking for a fault at $V_{control}$ = 0.78 percent * 6 V = 0.046875 V. It asserts the fault when (duty > **pid_osp_duty_thr**) AND (duty > FF_duty * **pid_osp_ff_scale**). With R2 short, the VSEN measured output voltage stays at 0 V, and the control loop increases duty much faster than normal start-up ramp. The duty-cycle equals **pid_osp_duty_thr** 29.64 percent after 360 µs of start-up. The FF_duty at 360 µs should be = (360 µs/10 ms) * 3.3 V/6 V = 1.98 percent. The duty (29.64 percent) is larger than the 1.5 x FF_duty and PID OSP fault is asserted.



(a)
R2 short
pid_osp_duty_thr = 38 (29.64 percent),
pid_osp_ff_scale = 3
pid_osp_ff_thr = 1

(b)
R1 short
pid_osp_duty_thr = 38 (29.64 percent),
pid_osp_ff_scale = 3
pid_osp_ff_thr = 1

Figure 185 Voltage sense open/short fault protection waveforms

Please note that the duty-cycle refers to IC internal duty-cycle, the PWM output duty-cycle is the internal duty-cycle less the dead-time if the dead-time applies to the rising edge, and plus the dead-time if the dead-time applies to the falling edge. In this example, the measured PWM is the primary clamp FET, its dead-time is 100 ns added to the falling edge, and thus the internal pulse width is 100 ns less of the probed waveform. The switching frequency is 300 kHz and the period is 3.3 µs. The XDPP1100 asserts PID OSP fault when PWM output pulse is wider than [(3.33 µs * 29.64 percent) + 100 ns] = 1.08 µs. The primary control FET pulse width is 200 ns shorter than the Clamp FET. The actual duty-cycle is (1.08 µs - 0.2 µs)/3.33 µs = 26.4 percent. The output voltage at the OSP detection is 26.4 percent * 6 V = 1.58 V. The calculated value matches the bench test result.

In R1 short situation, **Figure 185** (b), the output voltage is regulated at 0.8 V (VOUT_COMMAND = 3.3 V, VOUT_SCALE_LOOP = 0.2432). No fault is reported. The V_{OUT} is undervoltage and won't damage the board or the downstream circuit.

When the PID OSP common fault is triggered, the common fault bit [4] of PMBus command STATUS_MFR_SPECIFIC will be set to 1. The common fault list is shown in Table 106. The common fault status can be checked by writing 23_D to PMBus command 0xFE MFR_FIRMWARE_COMMAND, or selecting the "READ_COMMON_FAULTS_STATUS" from the drop-down list of the command. Then read the result from PMBus command 0xFD MFR_FIRMWARE_COMMAND_DATA.

For example, if the VSEN PID OSP fault is tripped, the common fault bit of the STATUS tab (**Figure 177**) will turn to red. Write 23_D to PMBus command 0xFE MFR_FIRMWARE_COMMAND and read 0xFD MFR_FIRMWARE_COMMAND_DATA should return = 00 00 10 00_H.

The response to a common fault can be set to either ignore or shut down. This is configured by the common fault shutdown mask register **fault_shut_mask_com** by setting the corresponding bit. For example, to enable VSEN PID OSP fault shutdown, write 4096_D to **fault_shut_mask_com**.

12.10 CML fault

The CML fault reports the data communication and memory faults.

Table 115 STATUS_CML data byte

Bit	Meaning
7	Invalid or unsupported command received
6	Invalid or unsupported data received
5	Packet error check failed
4	Memory fault detected
3	Processor fault detected
2	Not used
1	Other communication fault
0	Not used

A few examples of the CML fault.

- A MFR PMBus command is defined in a FW patch. A read or write action to this MFR command without the correct FW patch will report the "invalid/unsupported command" fault.
- A register has a range of values. Writing a number higher than the maximum value will report the "invalid/unsupported data" fault.
- The blank XDPP1100 without configuration stored in the OTP will report the "memory fault".
- A hard fault from the configurator will report the "processor fault". The hard fault could be:
  - Missing trim
  - Invalid partition size
  - Invalid partition base

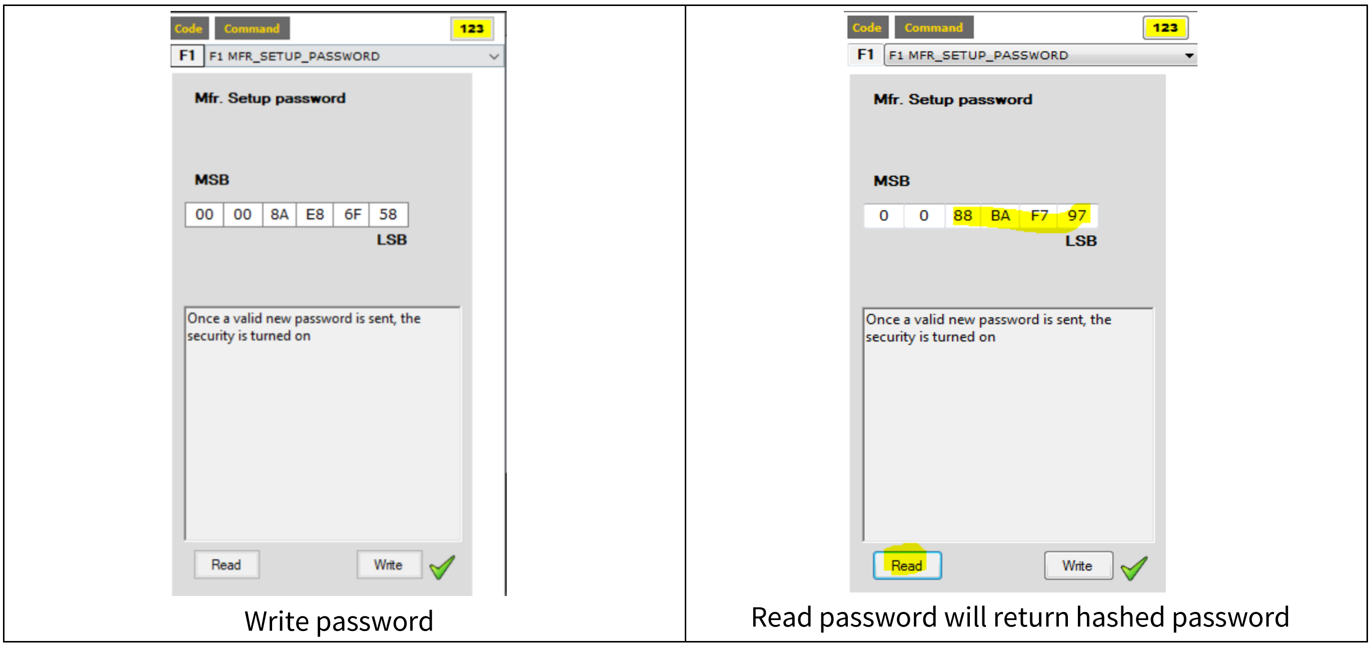
  - Missing configuration
  - Mis-match in PMBus data relevant to commands supported (data stored in OTP has fewer PMBus commands than patch supports)
  - OTP CRC error
- Packet Error Checking (PEC): whenever a PMBus device detects that the received and calculated PEC bytes do not match, the XDPP1100 will NACK the PEC byte and respond as follows:
  - Not respond to or act upon the received command
  - Flush or ignore the received command code and any received data
  - Set the CML bit in the STATUS_BYTE
  - Set the packet error check failed bit in the STATUS_CML register

12.11 PMBus command protection

The XDPP1100 provides a PMBus protection feature, which prevents the selected PMBus commands from being accidentally modified by an unauthorized user.

12.11.1 0xF1 MFR_SETUP_PASSWORD

The user can set up a password with PMBus command 0xF1 MFR_SETUP_PASSWORD to protect PMBus commands defined by 0xF4 and 0xF5. The length of the password can vary by user (up to 6 bytes). Read command 0xF1 will return a hashed password (**Figure 186**). Please note that only the last 4 bytes will be hashed; the two MSBs reveal the two MSBs of the password. Thus, it is recommended only use the last 4 bytes to set the user password.



Mfr. Setup password

MSB

00	00	8A	E8	6F	58

LSB

Once a valid new password is sent, the security is turned on

Write password

Mfr. Setup password

MSB

0	0	88	BA	F7	97

LSB

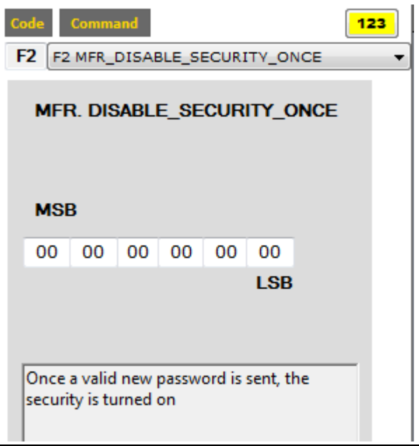
Once a valid new password is sent, the security is turned on

Read password will return hashed password

Figure 186 MFR_SETUP_PASSWORD

12.11.2 0xF2 MFR_DISABLE_SECURITY_ONCE

This is the PMBus command for password entry to disable password protection. It has a limit of four attempts before it is locked. A read returns: 0x000000000000 if security is off; 0x000000000001 if security is on; 0x000000000002 if security setup is locked due to incorrect password entry (**Figure 187**).



F2 F2 MFR_DISABLE_SECURITY_ONCE

MFR. DISABLE_SECURITY_ONCE

MSB

00	00	00	00	00	00

LSB

Once a valid new password is sent, the security is turned on

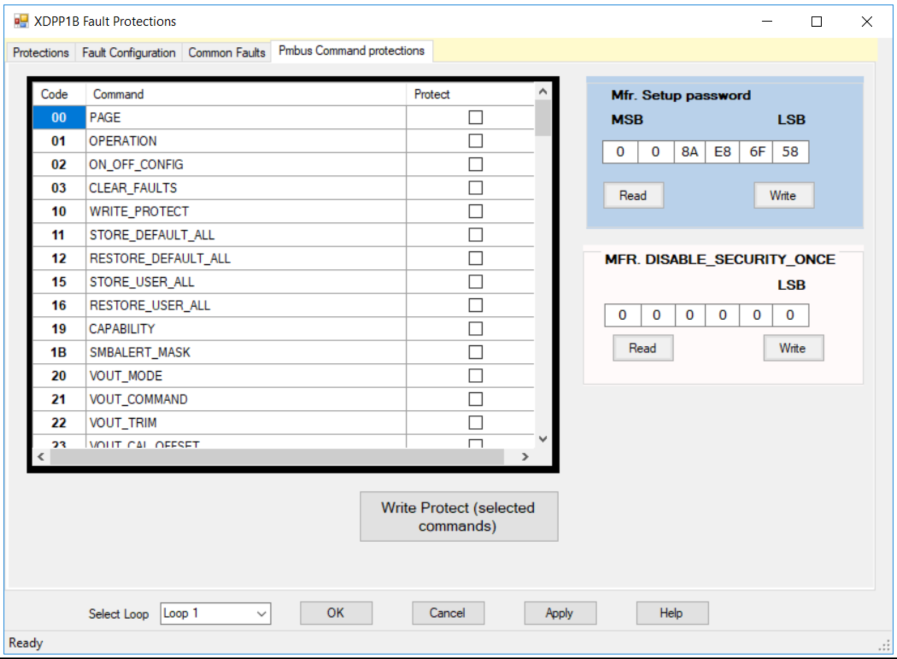
Figure 187 MFR_DISABLE_SECURITY_ONCE

Read MFR_DISABLE_SECURITY_ONCE, if it returns 0x000000000000, the user can go to MFR_SETUP_PASSWORD to set up a new password. If it returns 0x000000000001, the user has to write the previous set password here to turn-off the security. Read MFR_DISABLE_SECURITY_ONCE after writing the unlock password; it should return 0x000000000000 if the password is correct.

12.11.3 MFR_SECURITY_BIT_MASK

Once the security is turned on, the user can use commands 0xF4 MFR_SECURITY_BIT_MASK_LOW and 0xF5 MFR_SECURITY_BIT_MASK_HIGH to set the security mask of 255 PMBus commands.

0xF4 MFR_SECURITY_BIT_MASK_LOW masks bits for command numbers 0 to 127, 0xF5 MFR_SECURITY_BIT_MASK_HIGH masks bits for command numbers 128 to 255. A "1" in bit position is protected, "0" is unprotected. The commands under protection will not allow the user to write. To simplify the protection setup, the GUI offers a graphic tool to set the security bit mask. It can be accessed in the Faults & Protections design tool within the "PMBus Command protections" tab (**Figure 188**).



Code	Command	Protect
00	PAGE	□
01	OPERATION	□
02	ON_OFF_CONFIG	□
03	CLEAR_FAULTS	□
10	WRITE_PROTECT	□
11	STORE_DEFAULT_ALL	□
12	RESTORE_DEFAULT_ALL	□
15	STORE_USER_ALL	□
16	RESTORE_USER_ALL	□
19	CAPABILITY	□
1B	SMBALERT_MASK	□
20	VOUT_MODE	□
21	VOUT_COMMAND	□
22	VOUT_TRIM	□
23	VOUT_CAL_OFFSET	□

Mfr. Setup password

MSB LSB

0	0	8A	E8	6F	58

Read Write

MFR. DISABLE_SECURITY_ONCE

LSB

0	0	0	0	0	0

Read Write

Select Loop Loop 1

Figure 188 GUI design tool – "PMBus Command protections"

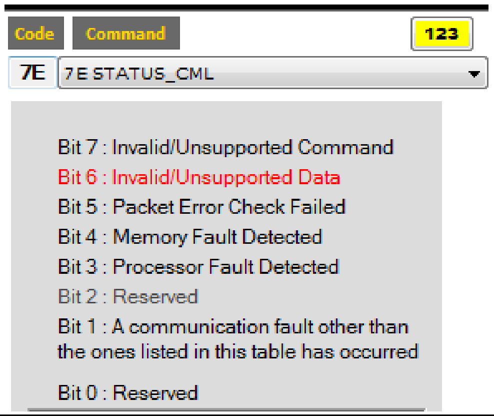
The “PMBus Command protections” tool enables the user to do the following:

- Check if the password is set by reading MFR_DISABLE_SECURITY_ONCE
- Write new password if the MFR_DISABLE_SECURITY_ONCE reads 00 00 00 00 00 00
- Select the PMBus that needs to be protected
- Set MFR_SECURITY_BIT_MASK by clicking the “Write Protect” button

In the case that user adds MFR PMBus commands through FW patch. Since the patched PMBus command is unpredictable, it is not included in the “PMBus command protections” design tool (**Figure 188**). User still can protect the patched command by directly write the bit through command 0xF4 and 0xF5. For example, if want to protect 0x3A FAN_CONFIG_1_2 command, write bit 59 of the MFR_SECURITY_BIT_MASK_LOW to 1 will protection 0x3A command.

If one command is selected for “Write Protect” and the security is on, re-write of this command is prohibited.

STATUS_CML command reports “Invalid/Unsupported Data” if trying to write a protected command (**Figure 189**).



Bit 7 : Invalid/Unsupported Command
Bit 6 : Invalid/Unsupported Data
Bit 5 : Packet Error Check Failed
Bit 4 : Memory Fault Detected
Bit 3 : Processor Fault Detected
Bit 2 : Reserved
Bit 1 : A communication fault other than the ones listed in this table has occurred
Bit 0 : Reserved

Figure 189 A CML fault is reported when a protected command attempts to be written

12.11.4 0x10 WRITE_PROTECT

The WRITE_PROTECT command offers the same function as the MFR password and security protection. The only difference is the WRITE_PROTECT doesn’t use password. The WRITE_PROTECT is also less flexibal, with only four options to choose from. The WRITE_PROTECT calls the same function in FW as the MFR password protection. Thus they are not independent and can’t use both. It is suggested only use one of the method to protect the PMBus commands; either using the MFR password and security bit mask, or the WRITE_PROTECT.

Table 116 WRITE_PROTECT command

Value	Meaning
0x00	Enable writes to all commands
0x20	Disable all writes except to the WRITE_PROTECT, OPERATION, ON_OFF_CONFIG, VOUT_COMMAND
0x40	Disable all writes except to the WRITE_PROTECT and OPERATION commands
0x80	Disable all writes except to the WRITE_PROTECT command

12.12 Protect I²C bus and register

The XDPP1100 could protect I²C registers by disabling the read/write of the entire I²C bus. While I²C bus is under protection, both write and read are prohibited. The protection is achieved by config PMBus command MFR_FIRMWARE_COMMAND_DATA and MFR_FIRMWARE_COMMAND. The instruction is shown below. User could further protect MFR_FIRMWARE_COMMAND_DATA and MFR_FIRMWARE_COMMAND using PMBus password protection.

Disable I2C bus

- Write 00 00 00 01 to command 0xFD MFR_FIRMWARE_COMMAND_DATA
- Write 0x0F to command 0xFE MFR_FIRMWARE_COMMAND or by selecting the “Disable_I2C_Bus” option form the drop-down list.

Enable I2C bus

- Write 00 00 00 00 to command 0xFD MFR_FIRMWARE_COMMAND_DATA
- Write 0x0F to command 0xFE MFR_FIRMWARE_COMMAND or by selecting the “Disable_I2C_Bus” option form the drop-down list.

13 GPIO and address offset

13.1 GPIO configuration

The XDPP1100 supports up to 16 GPIO pins in the 40-pin package and 11 GPIO pins in the 24-pin package. Table 117 is the GPIO pin muxing table. It can be seen that all the PWM pins can be configured as GPIO pins. The commonly used features, such as enable, power good and sync, are assigned to dedicated pins as their primary function. The function of a GPIO pin can be changed. For example, change the function of the MP_BEN (enable pin of the loop B) to a GPIO pin, or change the function of the SYNC pin to FAN1 PWM control.

Table 117 GPIO multipurpose pin

Name	Function 0	Function 1	Function 2	Function 3	Function 4	Function 5	Function 6	Function 7
MP_FAULT1	IO:GPIO0[2] (FAULT1)	IO:GPIO0[2]	IO:GPIO1[2]	IO:SYNC	O:FAN2_PWM	IO:SDA2	UARTRX	n/a
MP_FAULT2	IO:GPIO1[2] (FAULT2)	IO:GPIO0[2]	IO:GPIO1[2]	IO:SYNC	I:FAN2_TACH	IO:SCL2	UARTTX	n/a
MP_IMON	A:IMON	IO:GPIO0[3]	IO:GPIO1[3]	IO:SYNC	I:FAN1_TACH	n/a	n/a	n/a
PWM11	O:PWM11	IO:GPIO0[6]	IO:GPIO1[6]	IO:SYNC	O:FAN1_PWM	n/a	n/a	n/a
PWM12	O:PWM12	IO:GPIO0[7]	IO:GPIO1[7]	IO:SYNC	I:FAN1_TACH	n/a	n/a	n/a
PWM1	O:PWM1	IO:GPIO0[5]	IO:GPIO1[5]	IO:SYNC	n/a	n/a	n/a	n/a
PWM2	O:PWM2	IO:GPIO0[7]	IO:GPIO1[7]	IO:SYNC	n/a	n/a	n/a	n/a
PWM3	O:PWM3	IO:GPIO0[1]	IO:GPIO1[1]	IO:SYNC	n/a	n/a	n/a	n/a
PWM4	O:PWM4	IO:GPIO0[2]	IO:GPIO1[2]	IO:SYNC	n/a	n/a	n/a	n/a
PWM5	O:PWM5	IO:GPIO0[3]	IO:GPIO1[3]	IO:SYNC	UARTRX	n/a	n/a	n/a
PWM6	O:PWM6	IO:GPIO0[4]	IO:GPIO1[4]	IO:SYNC	UARTTX	n/a	n/a	n/a
PWM9	O:PWM9	IO:GPIO0[4]	IO:GPIO1[4]	IO:SYNC	n/a	n/a	n/a	n/a
PWM10	O:PWM10	IO:GPIO0[5]	IO:GPIO1[5]	IO:SYNC	n/a	n/a	n/a	n/a
PWM7	O:PWM7	IO:GPIO0[5]	IO:GPIO1[5]	IO:SYNC	O:FAN2_PWM	n/a	n/a	n/a
PWM8	O:PWM8	IO:GPIO0[6]	IO:GPIO1[6]	IO:SYNC	I:FAN2_TACH	n/a	n/a	n/a
MP_SMBALERT#	IO:SMBALERT_N	IO:GPIO0[6]	IO:GPIO1[6]	IO:SYNC	IO:GPIO0[7]	IO:GPIO1[7]	n/a	n/a
MP_BEN	IO:GPIO1[0] (BEN)	IO:GPIO0[0]	IO:GPIO1[0]	IO:SYNC	UARTRX	IO:SDA2	n/a	n/a
MP_ BPWRGD	IO:GPIO1[1] (BPWRGD)	IO:GPIO0[1]	IO:GPIO1[1]	IO:SYNC	UARTTX	IO:SCL2	n/a	n/a
MP_SYNC	n/a	IO:GPIO0[7]	IO:GPIO1[7]	IO:SYNC	O:FAN1_PWM	n/a	n/a	n/a
MP_EN	IO:GPIO0[0](EN)	IO:GPIO0[0]	IO:GPIO1[0]	IO:SYNC	n/a	n/a	n/a	n/a
MP_PWRGD	IO:GPIO0[1] (PWRGD)	IO:GPIO0[1]	IO:GPIO1[1]	IO:SYNC	n/a	n/a	n/a	n/a

The function of each GPIO pin is configured by register xxxx_func. For example, the power good PWRGD pin is configured by **pwrgd_func** register, see Table 118. The function of each GPIO pin is listed in Table 117. The output of the GPIO pin can be configured to CMOS output (push-pull) or open-drain output. Each GPIO pin has an internal weak pull-up and weak pull-down resistor. The weak pull-up and weak pull-down can be disabled.

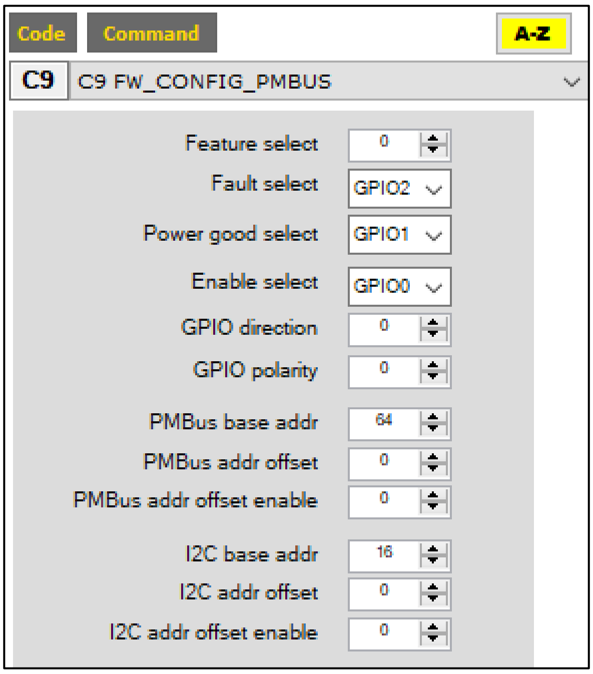
Table 118 GPIO function configuration (PWRGD example)

Register name	Description
pwrgd_func	Pin PWRGD function definition. As in Table 117, it can be configured as follows: 0 = PWRGD 1 = GPIO0[1] 2 = GPIO1[1] 3 = SYNC
pwrgd_pd	Pin PWRGD weak pull-down enable. 0 = pull-down disabled (recommended) 1 = pull-down enabled
pwrgd_pu_n	Pin PWRGD weak pull-up enable.

	0 = pull-up enabled 1 = pull-up disabled (recommended)
pwrgd_ppen	Pin PWRGD output buffer CMOS/open drain select. 0 = open drain output 1 = CMOS output

The polarity of GPIO pins is configured by PMBus command 0xC9 FW_CONFIG_PMBUS (**Figure 190**). The user can use this command to do the following configuration.

- Configure three common GPIO pins: Power good, enable and fault. The default setting is shown in **Figure 190**.
- Set GPIO polarity. Polarity is only set on initialization, so it would have to be stored in OTP and will take effect after power cycling the 3.3 V V_{DD} or resetting FW. Polarity should go by bit position: enable = bit[0], power good = bit[1], fault = bit[2]. Loop 0 FW_CONFIG_PMBUS configures GPIO0[x], loop 1 configures GPIO1[x]. Here the x value is 0 to 7. For example, if we want to set power good polarity GPIO0[1] to "active high", writing a value of 2 to the GPIO polarity field will set bit [1].
- Set PMBus base address. For example, set to 64 (0x40); this is the default setting and the user could change it to another value.
- Set I^2C base address. For example, set to 16 (0x10), this is the default setting and user could change it to another value. Please note that the PMBus address and I^2C address can't be set to the same.
- To enable XADDR1, XADDR2 resistor to program address offset, set "PMBus addr offset enable" = 1, set "I2C addr offset enable" = 1. Otherwise, set these two bits to 0.
- The "GPIO direction" and "Feature select" are not used and are reserved for future patching.



C9 | C9 FW_CONFIG_PMBUS

Feature select	0
Fault select	GPIO2
Power good select	GPIO1
Enable select	GPIO0
GPIO direction	0
GPIO polarity	0
PMBus base addr	64
PMBus addr offset	0
PMBus addr offset enable	0
I2C base addr	16
I2C addr offset	0
I2C addr offset enable	0

Figure 190 FW_CONFIG_PMBUS

The polarity of the enable pin can also be configured by PMBus command 0x02 ON_OFF_CONFIG. The enable pin polarity is set by bit [1]. A drop-down list is offered by the GUI for the user to select "active high" or "active low".

Table 119 explains the meaning of FW_CONFIG_PMBUS in detail.

Table 119 FW_CONFIG_PMBUS data byte format

Bit(s)	Parameter	Meaning	Example (loop 0)
56:54	fault_select	GPIO bit to map to fault	Default 2: map GPIO0[2] to Fault1
53:51	power_good_select	GPIO bit to map to power good	Default 1: map GPIO0[1] to PWRGD
50:48	enable_select	GPIO bit to map to enable	Default 0: map GPIO0[0] to EN
47:40	gpio_direction	Direction of given pin	Define input or output (not used)
39:32	gpio_polarity	Polarity of PMBus GPIO driver	Define active low or high, set to 1 for active high
31:24	pmbus_addr	Base address for PMBus offset to start from	Default: 64 (x40)
23:17	pmbus_addr_offset	PMBus address offset when resistor offset not enabled	
16	pmbus_addr_offset_en	Enable PMBus address offset via resistor	
15:8	i2c_addr	Base address for I^2C offset to start from	Default: 16 (x10)
7:1	i2c_addr_offset	I^2C address offset when resistor offset not enabled	
0	i2c_addr_offset_en	Enable I^2C address offset via resistor	

13.2 Address offset configuration

As explained in the previous chapter, bit 0 and bit 16 of FW_CONFIG_PMBUS should be set to 1 to enable XADDR1 resistor offset. The configuration must be stored in OTP to take effect. This is because XDPP1100 only checks XADDR resistor offset at IC power-up.

The base address of PMBus and I^2C should be set to different values to avoid conflict. Once the configuration is stored in OTP, recycle 3.3 V V_{DD} and use the auto-populate function of the GUI to find the device. The GUI will scan the I^2C and PMBus address and identify the correct address offset defined by the FW_CONFIG_PMBUS.

The XDPP1100 supports 16-valent or 8-valent address offset. The address is configured by register **xv_decode_sel**; set to 0 will choose an 8-valent table, set to 1 will choose a 16-valent table. The address offset decoding table is shown below.

Table 120 I^2C/PMBus address offset of eight-segment decode

Resistor-to-GND (1 percent accuracy)		Address offset
XADDR1, or XADDR2	1.20 kΩ	0x07
	1.80 kΩ	0x06
	2.70 kΩ	0x05
	3.90 kΩ	0x04
	6.80 kΩ	0x03
	10.00 kΩ	0x02
	18.00 kΩ	0x01
	47.00 kΩ (or open)	0x00

Table 121 I²C/PMBus address offset of 16-segment decode

Resistor-to-GND (1 percent accuracy)		Address offset
XADDR1, or XADDR2	780 Ω	0x0F
	1.10 kΩ	0x0E
	1.50 kΩ	0x0D
	2.02 kΩ	0x0C
	2.70 kΩ	0x0B
	3.52 kΩ	0x0A
	4.70 kΩ	0x09
	6.07 kΩ	0x08
	8.00 kΩ	0x07
	10.20 kΩ	0x06
	13.20 kΩ	0x05
	17.20 kΩ	0x04
	22.47 kΩ	0x03
	29.20 kΩ	0x02
	39.00 kΩ	0x01
	56.00 kΩ (or open)	0x00

If a different address resistor lookup table is preferred, the user can write a FW patch to customize the resistor table.

13.3 GPIO de-glitch

A HW de-glitch circuit is impelemented to the EN and BEN function. The de-glitch should be enabled by register **gpioX_dben** and the delay time is configured by **gpio_dly**.

The de-glitch enable register gpioX_dben is bit-assigned. So, each GPIO de-glitch could be enabled/disabled independently. In the XDPP1100, the de-glitch only works for bit[0], which is the EN and BEN. And the GUI only allows setting the gpioX_dben to 0 or 1 to configure bit [0].

The de-glitch time can be set by **gpio_dly** from 2 µs to 8 µs, with resolution 1 µs. A setting of 0 disables the de-glitch function.

14 Layout guidelines

In order to optimize voltage regulator performance, it is important to minimize the effects of PCB parasitics. The following layout techniques highlight important practices that should be incorporated into the layout process to optimize the PCB.

14.1 Component placement

Within the allotted implementation area, orient the switching components first. The switching components are the most critical because they carry large amounts of energy and tend to generate high levels of noise. Switching component placement should take into account power dissipation. Align the output inductors and MOSFETs so that space between the components is minimized.

Critical small-signal components including the V_{DD} and VD12 decoupling capacitors, ISEN resistors, TSEN capacitors, and voltage feedback RC filters should be placed near the controller.

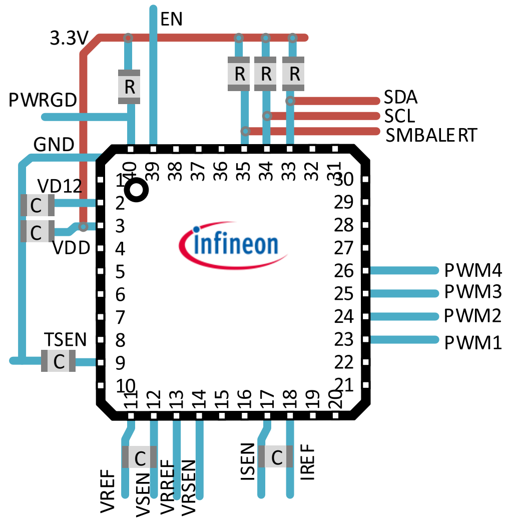


Figure 191 **Decoupling cap placement and routing**

14.2 Routing

For each voltage sense and current sense input, i.e. VSEN/VREF, ISEN/IREF, route the signal and its reference in differential pairs (Kelvin connection).

Avoid routing the VRSEN/VRREF, BVRSEN/BVRREF near any switching nodes, especially in dual-loop or two-phase applications. Unlike output voltage sensing, when VRSEN is configured to V_{RECT} sense mode, it measures the pulse signal. Thus, it cannot use a large filter to reduce noise. Keeping the trace shielded by the ground plane is recommended. Additional noise immunity is implemented in the IC.

14.2.1 Output current sense

Figure 192 is the layout example of output current sense. Avoid putting the current sense resistor or copper shunt next to any switching node. In high-gain current sense mode, put the XDPP1100 as close as possible to the sense resistor. One example of good practice is putting the XDPP1100 on top of the sense resistor on the other side of the PCB. If copper shunt is used for current sense, put the temperature-sense NTC or sense diode close to the copper shunt for accurate temperature compensation.

If low-gain mode is selected for current sense, put the current sense amplifier as close as possible to the shunt resistor.

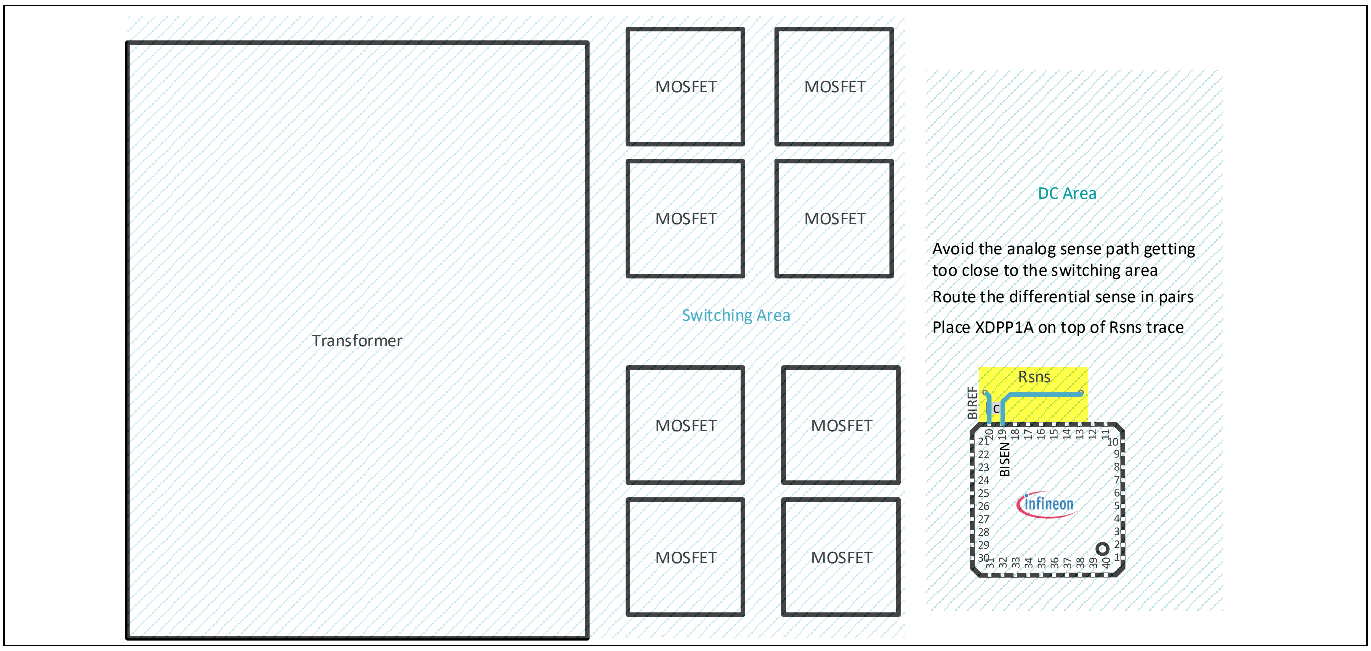

Transformer
MOSFET
MOSFET
MOSFET
MOSFET
Switching Area
MOSFET
MOSFET
MOSFET
MOSFET
DC Area
Avoid the analog sense path getting too close to the switching area
Route the differential sense in pairs
Place XDPP1A on top of Rsns trace
Rsns
BIREF
C
BISEN
infineon

Figure 192 Output current sense

14.2.2 Output voltage sense

Figure 193 is the layout example of output voltage sense. Avoid routing the voltage sense trace close to switching nodes. Always route the differential sense trace in pairs. Put the noise filter cap next to the IC pins.

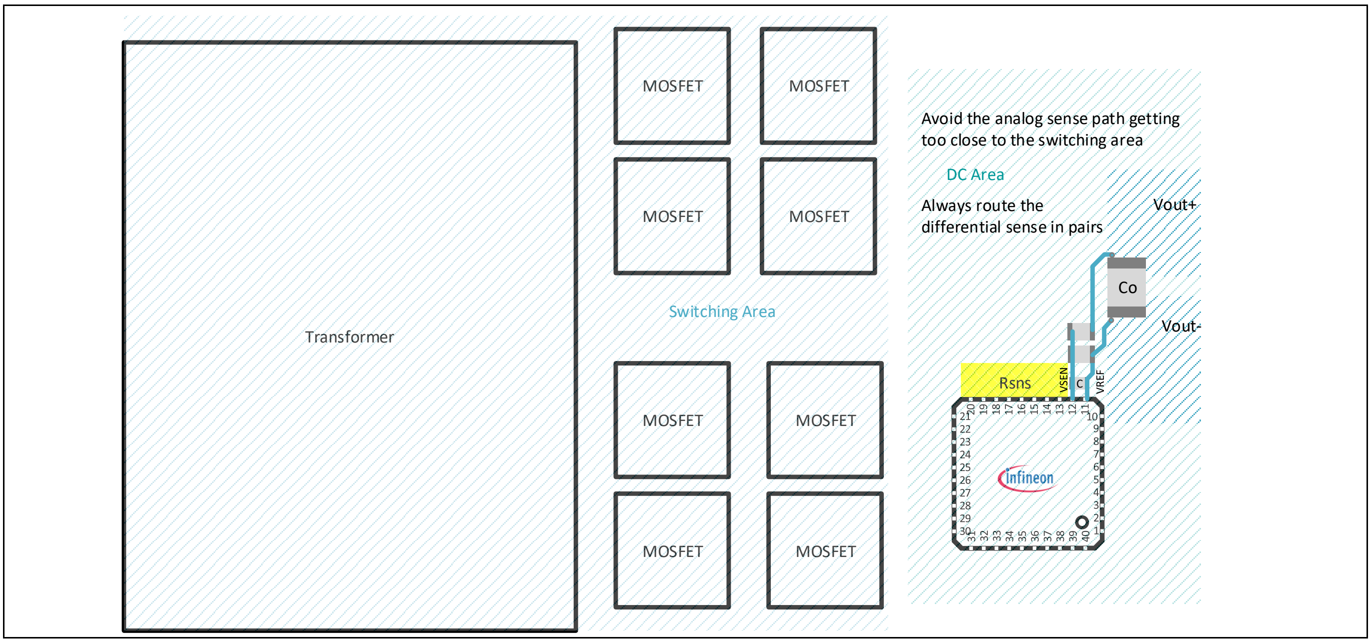


Figure 193 Output voltage sense

14.2.3 V_{RECT} voltage sense

Figure 194 is the layout example of V_{RECT} voltage sense. Route the differential sense trace in pairs. If VRSEN is used for flux balancing or adaptive dead-time adjustment, don’t put a filter cap on the VRSEN input; this will distort pulse shape and affect the accuracy of the time measurement.

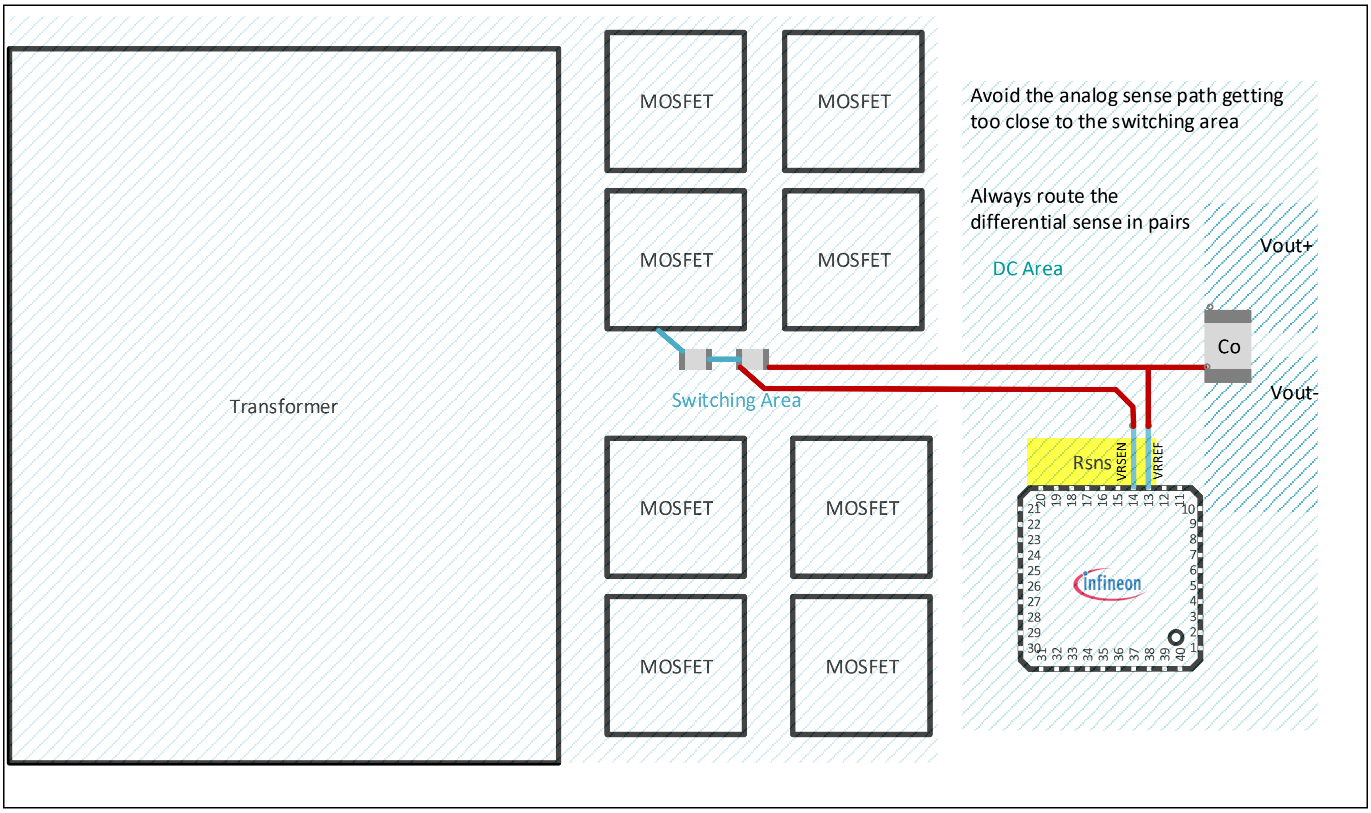


Figure 194 V_{RECT} voltage sense

14.2.4 GND connection

The XDPP1100 GND is the metal pad at the back side of the chip. Do not place too many vias to the GND plate in the layout. Use no more than 2 vias under the part. If the vias tie to internal ground layers, it is recommended to use a cross hatch pattern. This could avoid the internal ground planes pulling the heat away from upper pad during soldering.

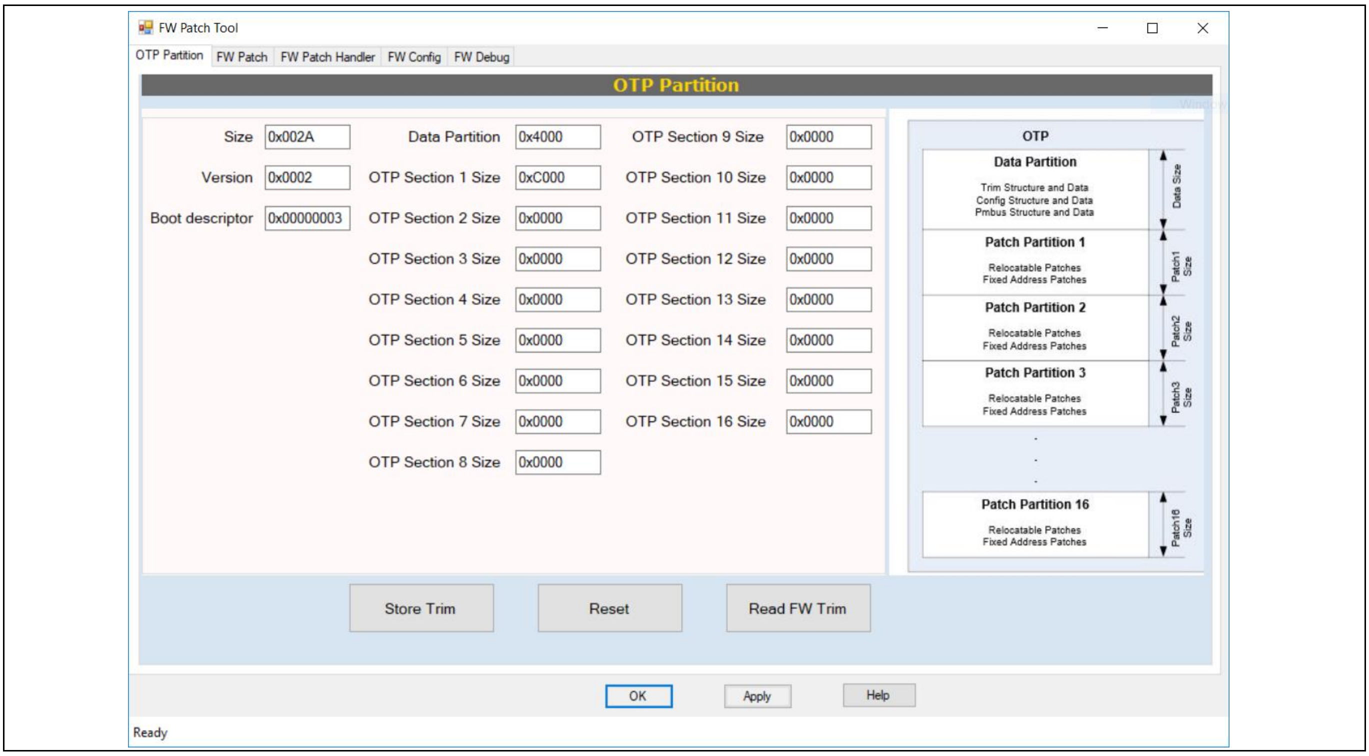
15 Store patch and configuration to OTP

Once the configuration or a FW patch is optimized and finalized, user should store the patch and configuration into OTP. The patch and configuration will be automatically loaded during the next boot up when IC is biased.

15.1 Check OTP partition

Before storing configuration, user should check if the OTP memory has been properly partitioned. In the FW Patch Tool > OTP partition tab, user could click "Read FW Trim" button to view the present OTP partition. By default, the OTP is partitioned to 16 kB data partition (0x4000) and 48 kB OTP partition 1 (0xC000). The data partition can be increased up to 31 kB if desired, the available OTP for FW patch will then be reduced to 33 kB given by the total 64 kB OTP memory. After changing partition size, click "Storm Trim" to save the partition info to OTP.

It is not possible to increase partition size if the next partition area is used. Take Figure 195 as an example, the data partition will not be configurable once a patch stored in OTP section 1.



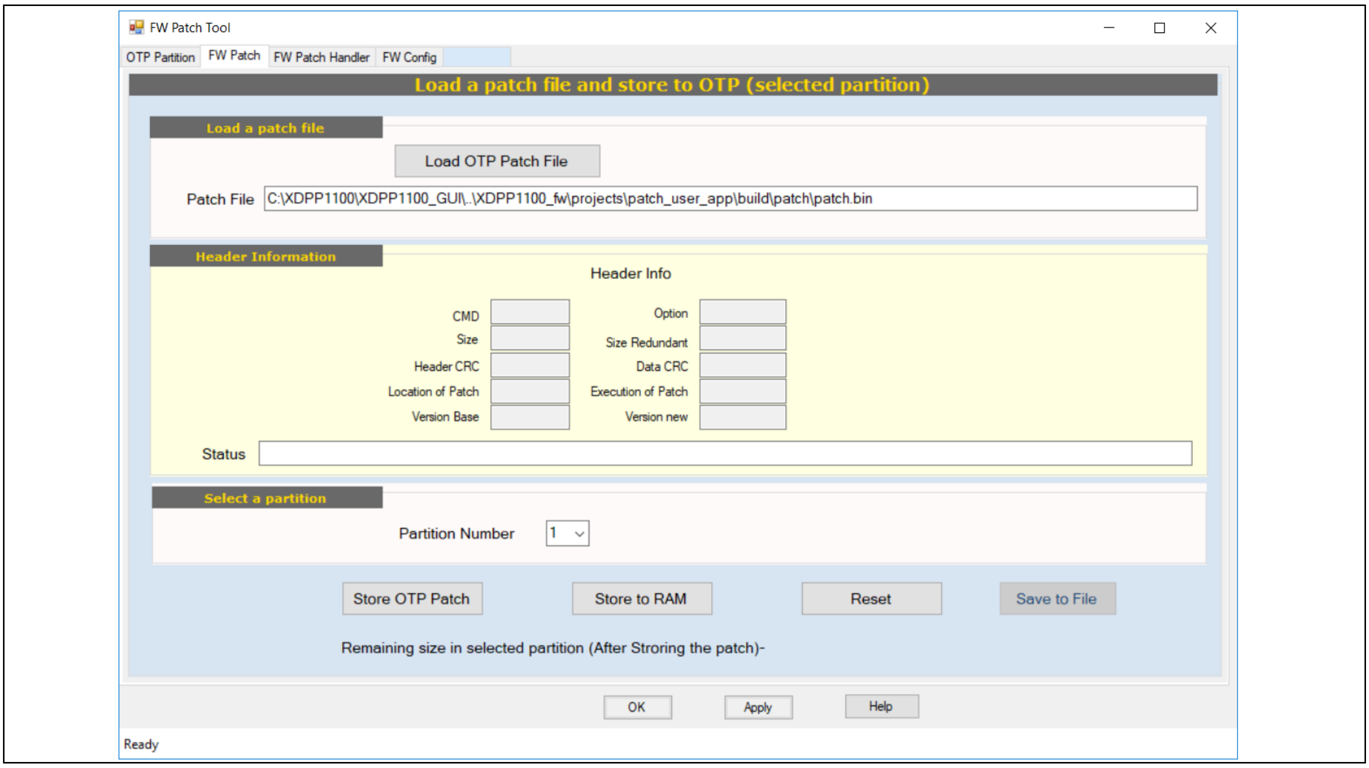
Size 0x002A
Version 0x0002
Boot descriptor 0x00000003
Data Partition 0x4000
OTP Section 1 Size 0xC000
OTP Section 2 Size 0x0000
OTP Section 3 Size 0x0000
OTP Section 4 Size 0x0000
OTP Section 5 Size 0x0000
OTP Section 6 Size 0x0000
OTP Section 7 Size 0x0000
OTP Section 8 Size 0x0000
OTP Section 9 Size 0x0000
OTP Section 10 Size 0x0000
OTP Section 11 Size 0x0000
OTP Section 12 Size 0x0000
OTP Section 13 Size 0x0000
OTP Section 14 Size 0x0000
OTP Section 15 Size 0x0000
OTP Section 16 Size 0x0000

Figure 195 OTP partition

15.2 Store FW patch

If a FW patch is developed, user could load and store the FW patch into OTP using the "FW Patch" design tool. Each OTP section an only have one active patch. The old patch must be invalidated before storing new patch to the same partition. The invalidation of existing patch is provided in GUI "FW patch handler" tool (Figure 197). The number of FW patch re-write depends on the size of the OTP section and the size of patch code.

To save OTP space, user should use RAM to verify the FW patch and configuration during debugging. There are 10 kB RAM available for customized FW patch and patch data. By default, the RAM is configured 8 kB for patch code and 2 kB for patch data. If the patch is larger than 8 kB (size > 0x2000), suggest break it into two smaller patches and verify the features separately. If the patch is larger than 8 kB and couldn't be broken, then store the patch into OTP. After a design is finalized, the patch code could also be stored in OTP.



FW Patch Tool

OTP Partition | FW Patch | FW Patch Handler | FW Config

Load a patch file and store to OTP (selected partition)

Load a patch file

Load OTP Patch File

Patch File: C:\XDPP1100\XDPP1100_GUI\..\XDPP1100_fw\projects\patch_user_app\build\patch\patch.bin

Header Information

Header Info

CMD, Size, Header CRC, Location of Patch, Version Base, Option, Size Redundant, Data CRC, Execution of Patch, Version new

Status

Select a partition

Partition Number: 1

Store OTP Patch | Store to RAM | Reset | Save to File

Remaining size in selected partition (After Stroring the patch)-

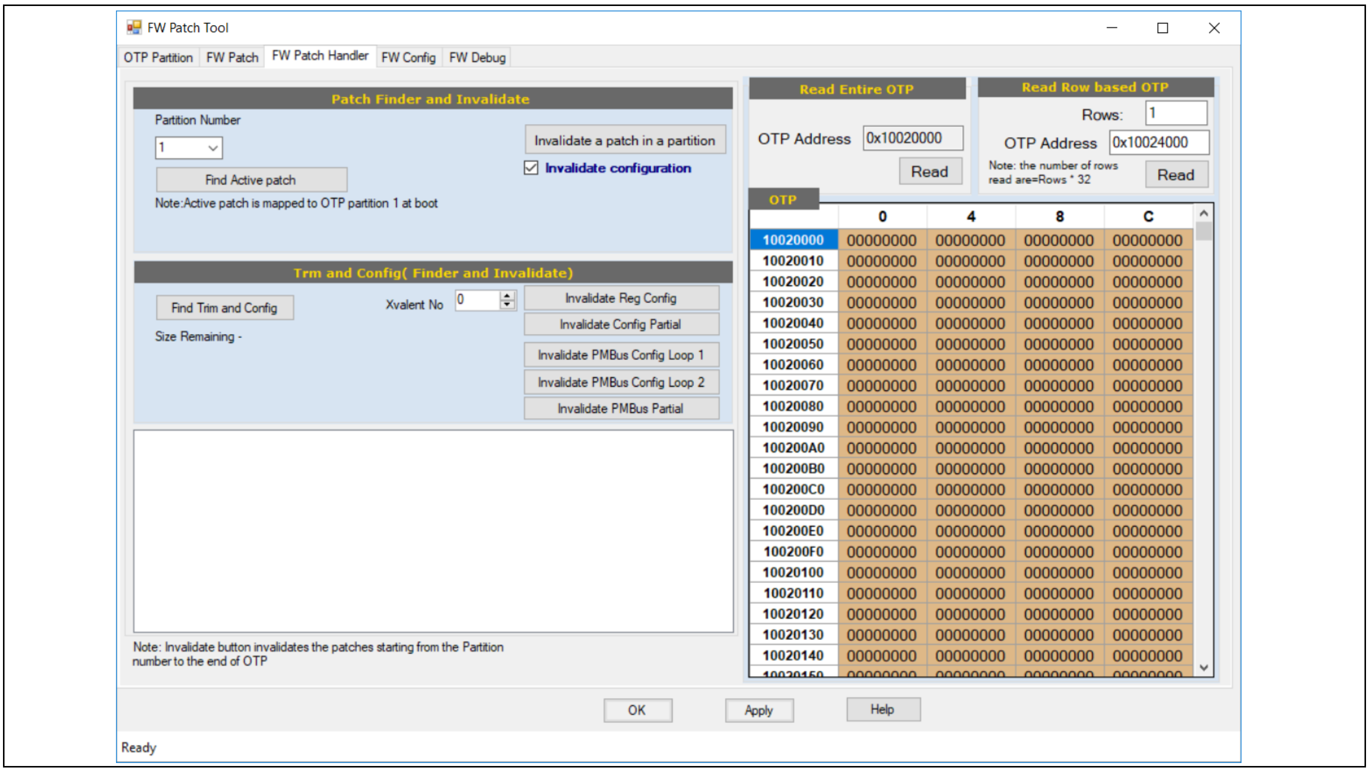
OK | Apply | Help

Ready

Figure 196 Load and store patch

15.3 FW patch handler

The XDPP1100 GUI FW patch handler offers find patch, invalidate patch, find configuration and invalidate configuration tools as shown in **Figure 197**.



FW Patch Tool

OTP Partition | FW Patch | FW Patch Handler | FW Config | FW Debug

Patch Finder and Invalidate

Partition Number: 1

Find Active patch

Note:Active patch is mapped to OTP partition 1 at boot

Invalidate a patch in a partition

☑ Invalidate configuration

Trm and Config(Finder and Invalidate)

Find Trim and Config

Xvalent No: 0

Size Remaining -

Invalidate Reg Config | Invalidate Config Partial | Invalidate PMBus Config Loop 1 | Invalidate PMBus Config Loop 2 | Invalidate PMBus Partial

Note: Invalidate button invalidates the patches starting from the Partition number to the end of OTP

Read Entire OTP

OTP Address: 0x10020000

Read

Read Row based OTP

Rows: 1

OTP Address: 0x10024000

Note: the number of rows read are=Rows * 32

Read

OTP

	0	4	8	C
10020000	00000000	00000000	00000000	00000000
10020010	00000000	00000000	00000000	00000000
10020020	00000000	00000000	00000000	00000000
10020030	00000000	00000000	00000000	00000000
10020040	00000000	00000000	00000000	00000000
10020050	00000000	00000000	00000000	00000000
10020060	00000000	00000000	00000000	00000000
10020070	00000000	00000000	00000000	00000000
10020080	00000000	00000000	00000000	00000000
10020090	00000000	00000000	00000000	00000000
100200A0	00000000	00000000	00000000	00000000
100200B0	00000000	00000000	00000000	00000000
100200C0	00000000	00000000	00000000	00000000
100200D0	00000000	00000000	00000000	00000000
100200E0	00000000	00000000	00000000	00000000
100200F0	00000000	00000000	00000000	00000000
10020100	00000000	00000000	00000000	00000000
10020110	00000000	00000000	00000000	00000000
10020120	00000000	00000000	00000000	00000000
10020130	00000000	00000000	00000000	00000000
10020140	00000000	00000000	00000000	00000000
10020150	00000000	00000000	00000000	00000000

OK | Apply | Help

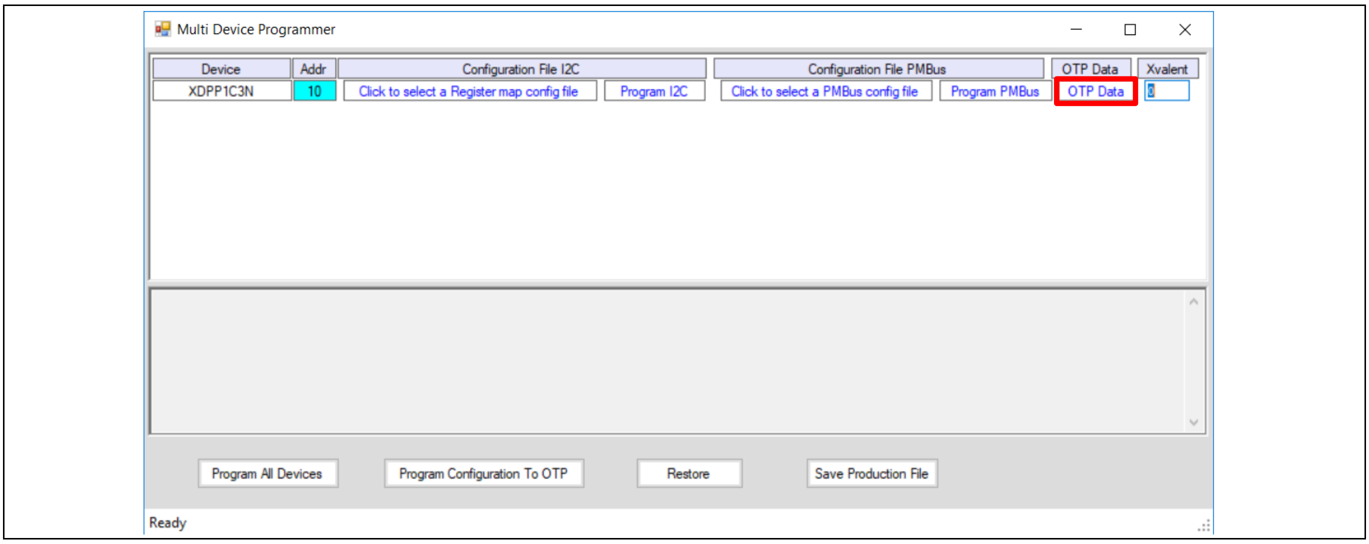
Ready

Figure 197 FW patch handler

15.4 Store configuration to OTP

The configuration should be loaded into RAM before storing into OTP. If this is not yet done, open the design file; click “write all (I2C and PMBus)” shortcut button to write all registers and PMBus commands to RAM.

Open the “Multi device programmer” by clicking the flame shortcut icon, set Xvalent number if you have multiple configuration to be stored in OTP. For single configuration, use the default Xvalent 0.

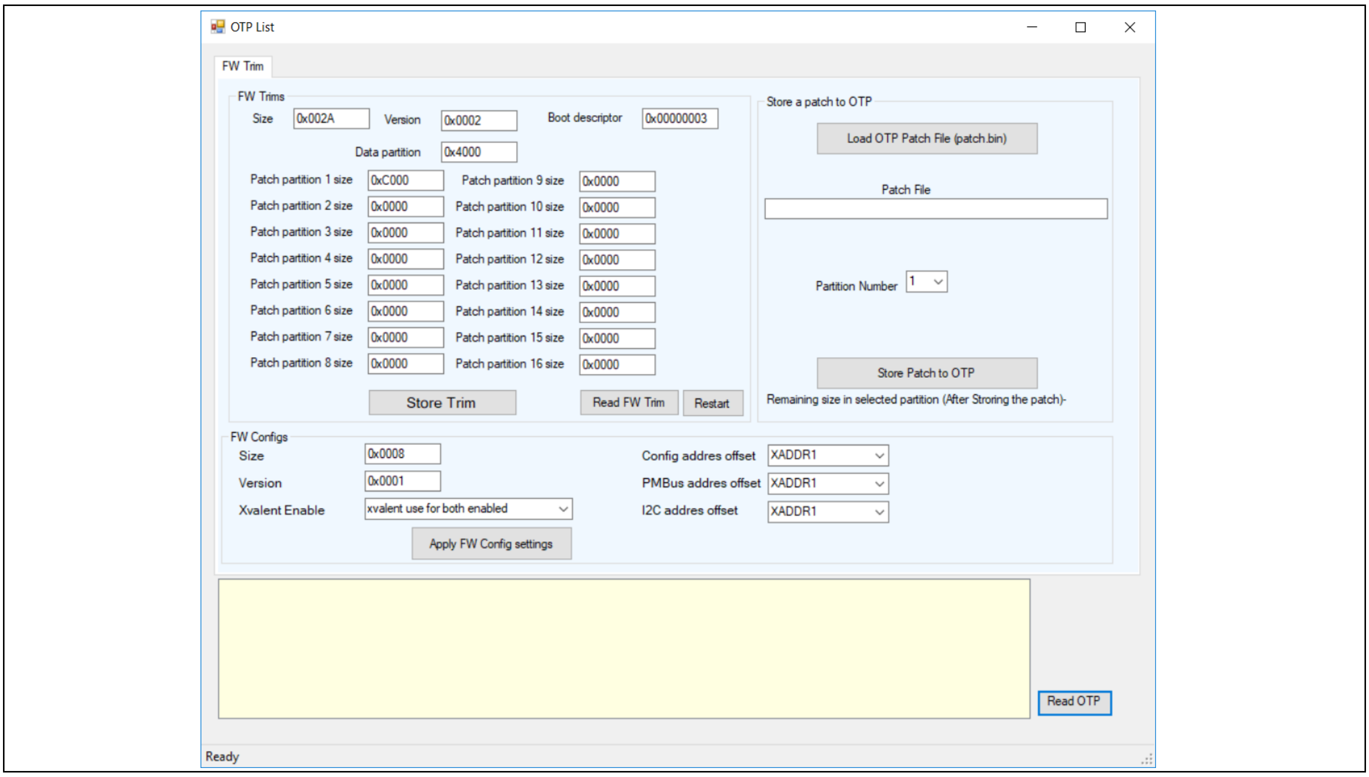


Multi Device Programmer

Device	Addr	Configuration File I2C		Configuration File PMBus		OTP Data	Xvalent
XDPP1C3N	10	Click to select a Register map config file	Program I2C	Click to select a PMBus config file	Program PMBus	OTP Data	0

Figure 198 Multi device programmer

Click the “OTP Data” could open a configuration window. In the FW config section, user could select which XADDR pin is going to be used for PMBus and I^{2}C address offset, and which XADDR pin for multi config offset (config address offset). Default configuration is XADDR1. Note only the 40-pin IC has XADDR2 pin.



OTP List

FW Trim

FW Trims

Size 0x002A Version 0x0002 Boot descriptor 0x00000003

Data partition 0x4000

Patch partition 1 size	0xC000	Patch partition 9 size	0x0000
Patch partition 2 size	0x0000	Patch partition 10 size	0x0000
Patch partition 3 size	0x0000	Patch partition 11 size	0x0000
Patch partition 4 size	0x0000	Patch partition 12 size	0x0000
Patch partition 5 size	0x0000	Patch partition 13 size	0x0000
Patch partition 6 size	0x0000	Patch partition 14 size	0x0000
Patch partition 7 size	0x0000	Patch partition 15 size	0x0000
Patch partition 8 size	0x0000	Patch partition 16 size	0x0000

Store a patch to OTP

Patch File

Partition Number 1

Remaining size in selected partition (After Stroring the patch)-

FW Configs

Size	0x0008	Config addres offset	XADDR1
Version	0x0001	PMBus addres offset	XADDR1
Xvalent Enable	xvalent use for both enabled	I2C addres offset	XADDR1

Figure 199 FW config

In the “Multi device programmer”, user could program I2C and PMBus in separate steps, or simply click the “Program configuration to OTP” button to store both I2C and PMBus configuration to OTP in one stop.

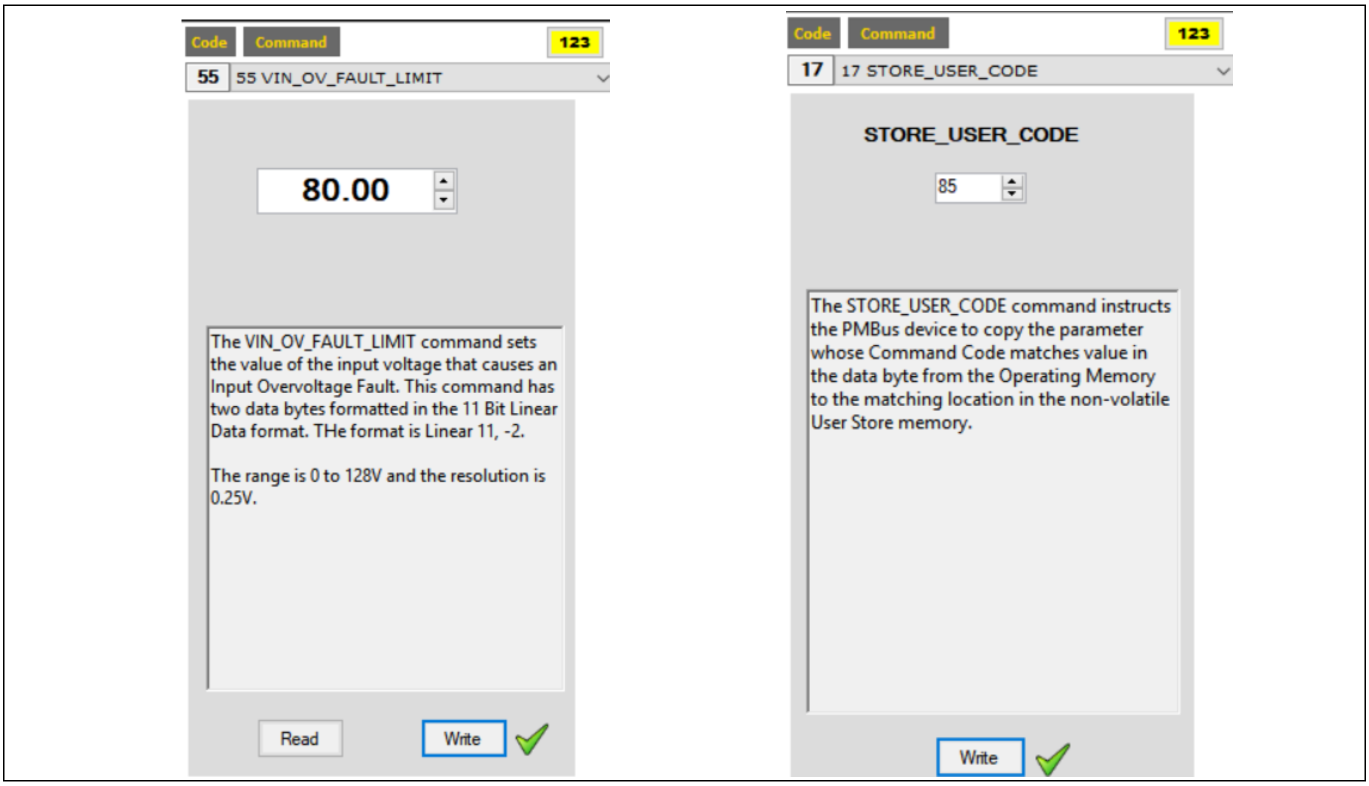
15.5 Partial config

If only a couple of parameters were modified after the whole configuration was stored in OTP, partial config could be used to update these parameters without re-write the whole configuration. This could save OTP space. Partial config can be PMBus command or registers config. Register partial config has no limitation on the number of stores (as long as OTP space is available) and PMBus partials are limited to 32.

Partial config can be stored into OTP while the chip is in regulation (on the fly update).

15.5.1 Store PMBus partial config

Use PMBus command 0x17 STORE_USER_CODE to write individual PMBus command to the OTP. For example, if the command 0x55 VIN_OV_FAULT_LIMIT was updated to a new value. Write the new threshold to 0x55 command. Then write 85 (hex 0x55 equals to decimal value 85), to 0x17 STORE_USER_CODE. The new VIN_OV_FAULT_LIMIT will be saved to OTP.



The VIN_OV_FAULT_LIMIT command sets the value of the input voltage that causes an Input Overvoltage Fault. This command has two data bytes formatted in the 11 Bit Linear Data format. THe format is Linear 11, -2.

The range is 0 to 128V and the resolution is 0.25V.

The STORE_USER_CODE command instructs the PMBus device to copy the parameter whose Command Code matches value in the data byte from the Operating Memory to the matching location in the non-volatile User Store memory.

Figure 200 STORE_USER_CODE

15.5.2 Store register partial config

In the register map, use the “Group Command” tool to store a register partial config. Double click the register that requires modification. It will be brought to the “group command” area. In the ValHex column, change register value. Please note, the value in “group command” is in hex format. For example, 35 in decimal is 0x23 in hex. In Figure 201, modify the value from 23 to 25 and click enter. Then click “Write Partials” would store the register partial config into OTP.

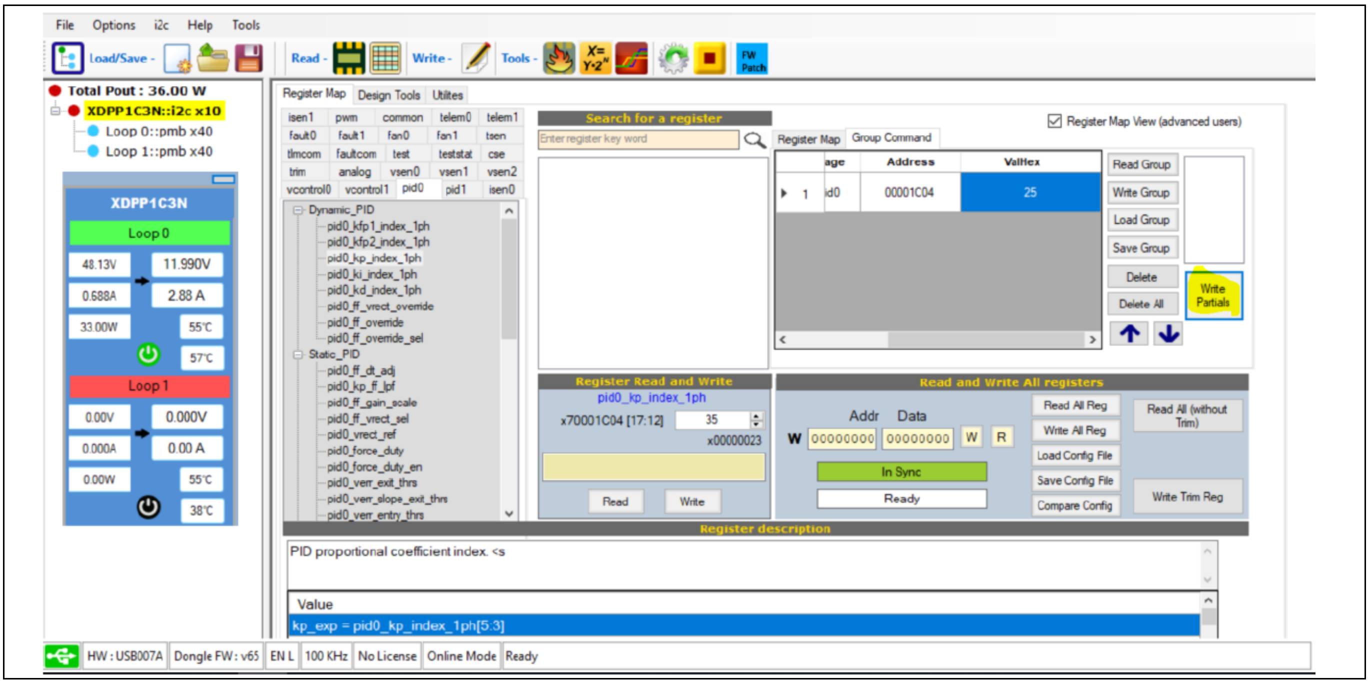


Figure 201 **Write register partial config**

Below are the steps to program an individual register data to OTP.

- Write Register address and register data value to scratch pad address 0x20061800 (RAM_SCRATCHPAD_ADDRESS) and 0x20061804 (RAM_SCRATCHPAD_ADDRESS + 4).

  Example: To store a partial config of register address 0x70000400, a value of 0x0000B409.

  We perform two I2c transactions to write the address and then data in consecutive location of scratch pad.

  [0x20] [0x00] [0x18] [0x06] [0x20] [0x00] [0x04] [0x00] [0x70]

  [0x20] [0x04] [0x18] [0x06] [0x20] [0x09] [0xB4] [0x00] [0x00]

- Write PMBus command 0xFD MFR_FIRMWARE_COMMAND_DATA with 4 Bytes, from LSB to MSB as - [Xvalent] [size low byte] [size high byte] [00]

  i.e. Byte0 = Xvalent number, Byte 1 = size low byte, Byte2 = size high byte, Byte3 = reserved

  [0x80] [0x00] [0x00] [0x80] [0xFD] [0x04] [0x00] [0x08] [0x00] [0x00]

- Write PMBus command 0xFE MFR_FIRMWARE_COMMAND with input argument - 0x14

  [0x80] [0x00] [0x00] [0x80] [0xFE] [0x14]

Below is the pseudo code to do this in program.

```
status = i2cWrite4Bytes(RAM_SCRATCHPAD_ADDRESS, “0x70000400”)
status1 = i2cWrite4Bytes(RAM_SCRATCHPAD_ADDRESS+4,“0x0000B409”)
        writeData1(0) = XvalentNumber
        writeData1(1) = length_Low_Byte
        writeData1(2) = length_High_Byte
        writeData1(3) = 0
```

```
status = pmbWriteBlock(deviceIndex,0,cmdMFR_FIRMWARE_COMMAND_DATA,
writeData1)
```

```
status = pmbWriteBytewithoutSavingBuffer(deviceIndex, 0,
cmdMFR_FIRMWARE_COMMAND, 20)
```

16 Nomenclature

Table 122 Definitions of acronyms, symbols and terms

Acronym, symbol or term	Definition
ACF	Active clamp forward
ADC	Analog-to-digital converter
BOM	Bill of materials
CCM	Continuous conduction mode
CRC	Cyclical redundancy checking
DAC	Digital-to-analog converter
DCM	Discontinuous conduction mode
DE	Diode emulation
FB-FB	Full-bridge full-bridge rectifier topology
FTR	Fast transient response
F_{SW}	Switching frequency of converter
FW	Firmware
GPIO	General purpose input/output
GUI	Graphical user interface
HB	Half-bridge
HBCT	Half-bridge center-tap topology
HiZ	High impedance
HW	Hardware
LPF	Low-pass filter
LSB	Least significant bit
N_p	Number of turns of the transformer primary winding
N_s	Number of turns of the transformer secondary winding
NTC	Negative temperature coefficient
OCP	Over-current protection
OSP	Open-sense protection
OTP	One-time programmable memory
OTP	Overtemperature protection
OVP	Overvoltage protection
PCB	Printed circuit board
PCL	Peak current limit
PCMC	Peak current mode control
PID	Proportional, integral, derivative coefficient
PI filter	Proportional integral filter
PTC	Positive temperature coefficient
PWM	Pulse width modulation

Acronym, symbol or term	Definition
$R_{DS(on)}$	MOSFET on-state resistance
SCP	Short-circuit protection
SR	Synchronous rectification
T_{SW}	Switching period of converter
UCP	Under-current protection
UTP	Under-temperature protection
UVP	Undervoltage protection
V_{IN}	Input voltage
VMC	Voltage mode control
V_{OUT}	Output voltage
V_{RECT}	Transformer secondary rectified voltage

[1] XDPP1100 datasheet

[2] XDPP1100 gui installation and user guide

[3] XDPP1100 product overview

[4] XDPP1100 technical reference manual

Revision history

Document version	Date of release	Description of changes
V 1.0	22-06-2020	First release
V1.1	26-10-2020	Chapter 3.2.1 updated IADC description, added tracking ADC step size information. Chapter 6.1.7 updated fbal_lpf_kpshift equation. Chapter 12.3.2 added description to fault_hyst. Corrected typo in Table 80 and Table 95. Update Table 102 added fault_temp_src_sel.
V1.2	2021-01-12	Add chapter 14.2.4, GND layout guideline.
	2021-04-12	Added MFR_IOUT_APC calculation in chapter 3.2.2.2. Updated description of burst mode, chapter 9. The burst mode only applies to bridge topologies. Added description in chapter 12.11.3, how to protect user added MFR PMBus command. Updated description of 12.11.4, The WRITE_PROTECT calls the same function in FW as the MFR password protection. Thus, they are not independent and can’t use both. Add chapter 12.12, protect i2c bus and register. Add chapter 15, store configuration to OTP, store partial config.

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

Edition 2021-04-23

Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Email: erratum@infineon.com

Document reference
AN_2003_PL88_2006_221026

IMPORTANT NOTICE
The information contained in this application note is given as a hint for the implementation of the product only and shall in no event be regarded as a description or warranty of a certain functionality, condition or quality of the product. Before implementation of the product, the recipient of this application note must verify any function and other technical information given herein in the real application. Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this application note.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office (**www.infineon.com**).

WARNINGS
Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.