RC38312, RC38112, RC38208, RC38108

FemtoClock®3 Radio Synchronizer and Multi-Frequency Clock Synthesizer

The RC38312, RC38112, RC38208, RC38108 are ultra-low phase noise timing devices with jitter attenuator, multi-frequency synthesizer, synchronous Ethernet synchronizer, and digitally controlled oscillator (DCO) capabilities (see the following figures). These flexible, low-power devices output clocks with ultra-low in-band phase noise and spurious for 4G and 5G RF transceivers and with jitter below 25fs-rms for 112Gbps and 224Gbps SerDes.

This document provides programming information for software engineers to use the RC38312, RC38112, RC38208, RC38108.

Note: Device specifications and ordering information are available in the datasheets for the respective devices.

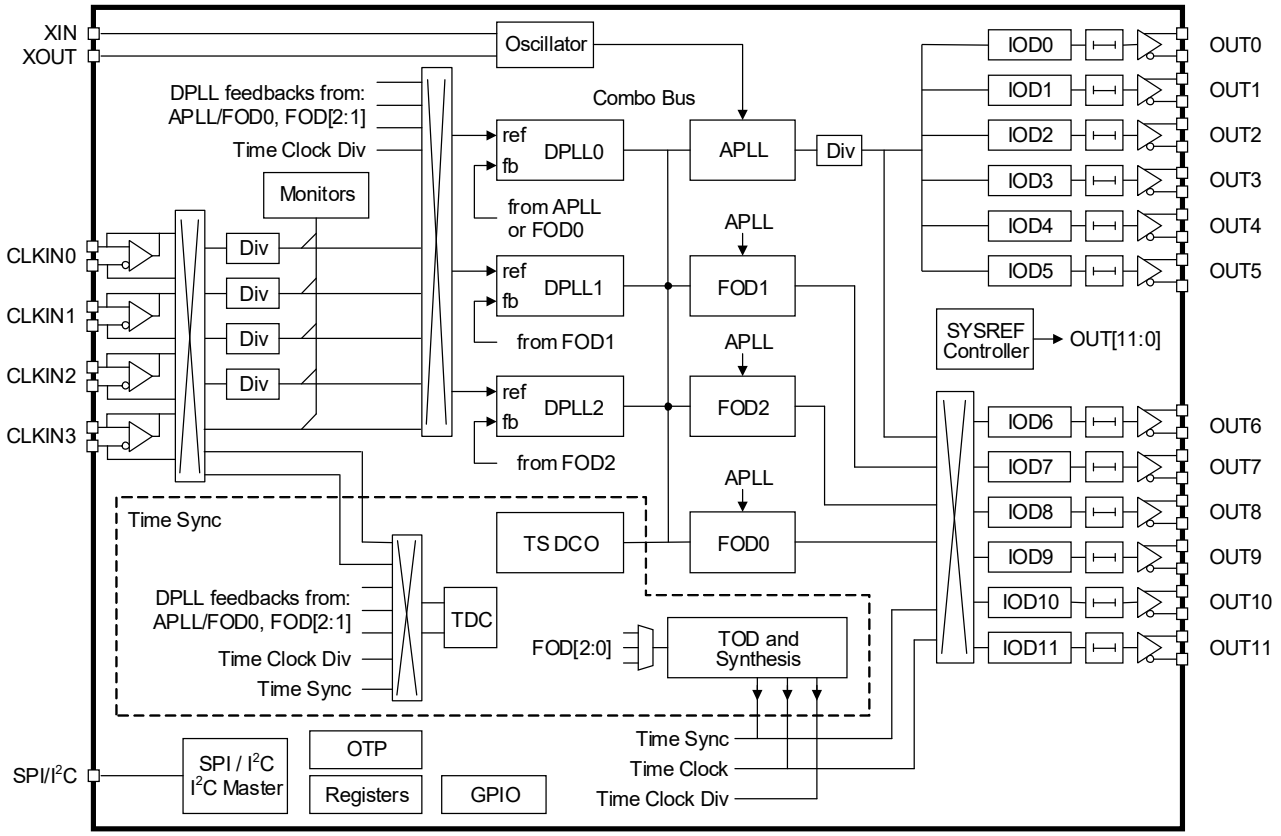


Figure 1. RC38312 Block Diagram

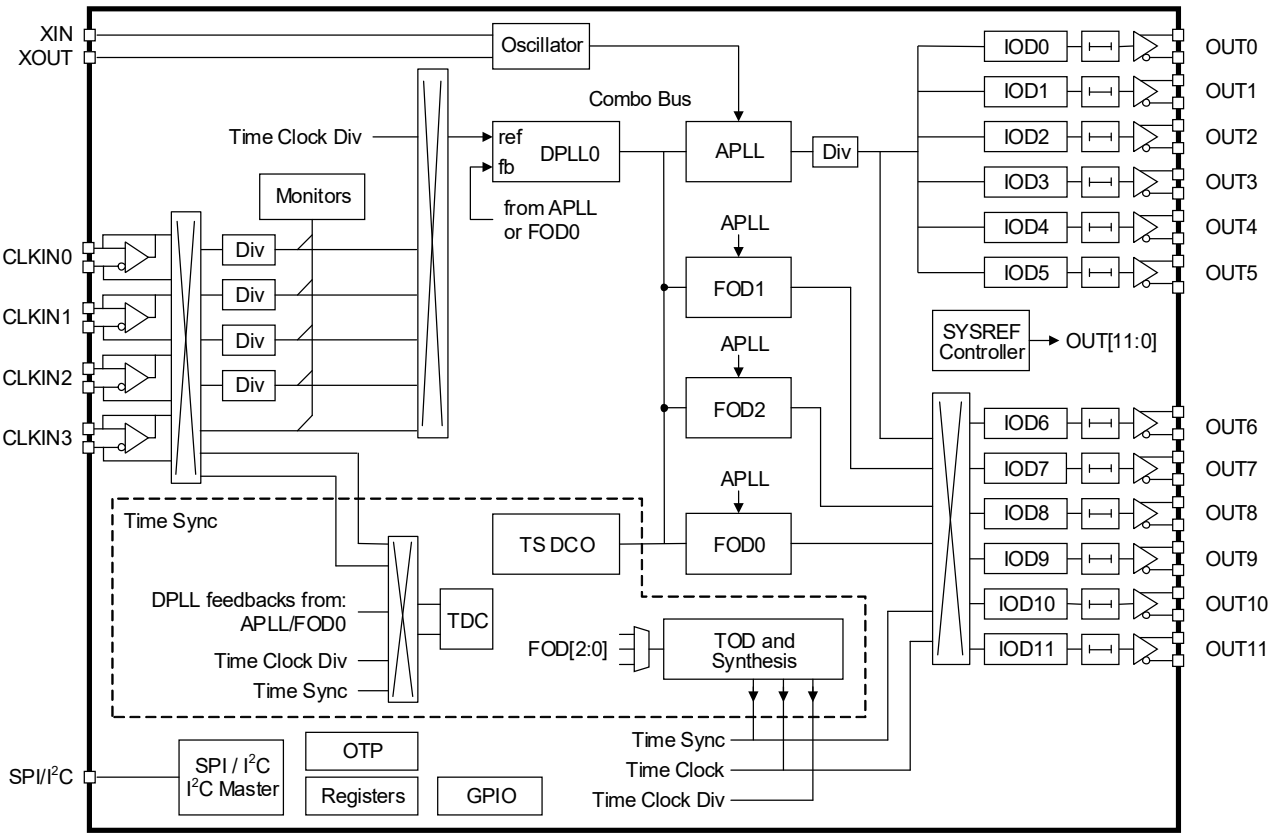


Figure 2. RC38112 Block Diagram

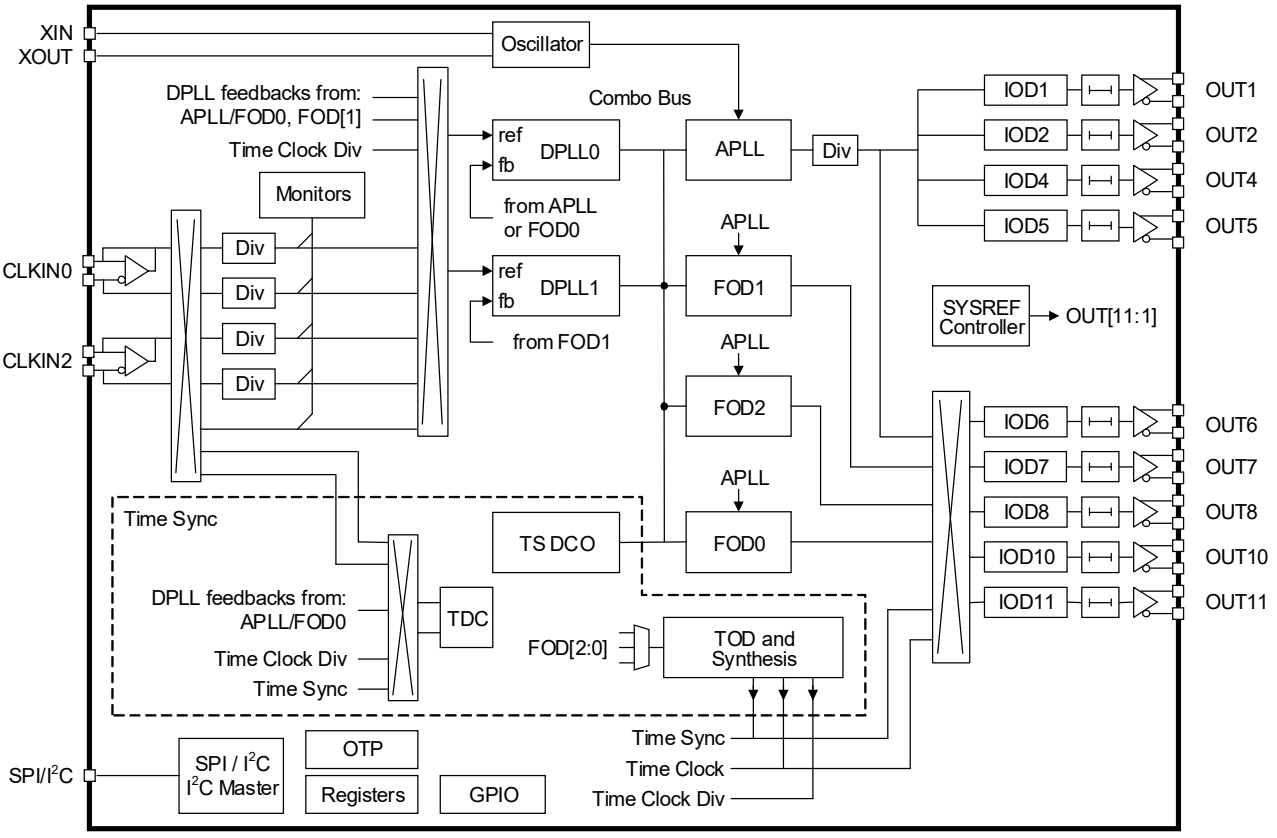


Figure 3. RC38208 Block Diagram

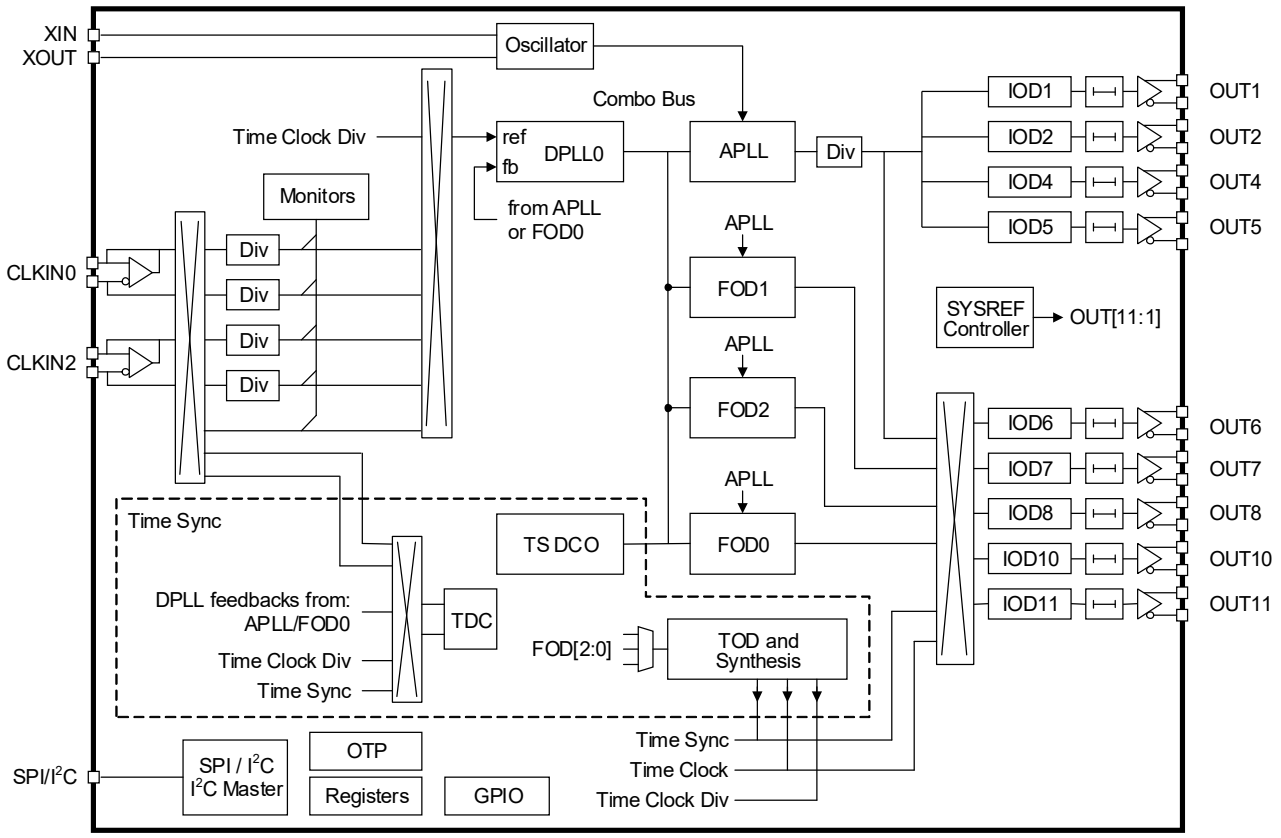


Figure 4. RC38108 Block Diagram

Contents

Figures

Tables

1. Functional Description

1.1 Overview

The RC38312, RC38112, RC38208, RC38108 are ultra-low phase noise timing devices with jitter attenuator, multi-frequency synthesizer, synchronous Ethernet synchronizer, and digitally controlled oscillator (DCO) capabilities. These flexible, low-power devices output clocks with 25fs RMS jitter supporting 112Gbps and 224Gbps SerDes.

All devices provide an ultra-low phase noise analog PLL (APLL) clock synthesizer and three fractional output divider (FOD) clock synthesizers. Differences between the devices are summarized in Table 1.

For all devices, the differential clock inputs can each be configured as two single-ended inputs. The clock inputs can operate at frequencies up to 1GHz when differential and 250MHz when single-ended.

Table 1. Comparison of Product Family Members

Part Number	Block Diagram	Number of DPLL Ch.	Number of CLKIN	Number of OUT	Package
RC38312	Figure 1	3	4	8 x LVDS / HCSL / CML 4 x LVDS / HCSL / LVCMOS	100-BGA
RC38112	Figure 2	1	4	8 x LVDS / HCSL / CML 4 x LVDS / HCSL / LVCMOS	100-BGA
RC38208	Figure 3	2	2	6 x LVDS / HCSL / CML 2 x LVDS / HCSL / LVCMOS	64-BGA
RC38108	Figure 4	1	2	6 x LVDS / HCSL / CML 2 x LVDS / HCSL / LVCMOS	64-BGA

For all devices, all present OUT[7:0] can be configured as LVDS, HCSL (AC-LVPECL) or CML. When configured for LVDS or HCSL, the differential outputs can operate at frequencies up to 1GHz. When configured for CML, the differential outputs can operate a frequencies up to 2.5GHz.

For all devices, all present OUT[11:8] can be configured as LVDS, HCSL (AC-LVPECL) or LVCMOS. When configured for LVDS or HCSL, the differential outputs can operate at frequencies up to 1GHz. Each differential output can be configured as two LVCMOS outputs that can operate at frequencies up to 250MHz.When configured for LVCMOS the OUT and nOUT pins can operate in-phase or 180° out of phase.

Figure 7 shows a detailed functional block diagram of RC38312. This diagram, with the appropriate functional blocks removed, is representative of the architectures of all the devices in the family.

1.2 Combo Bus

The RC38312, RC38112, RC38208, RC38108 includes four clock synthesizers, an APLL based synthesizer and 3 FOD based synthesizers. The fractional frequency offset (FFO) of each synthesizer can be steered by the DPLL and DCO channels within the device. The DPLL and DCO channels output frequency steering words that are distributed to the synthesizers via the Combo Bus, please see Figure 9, Figure 10, and Figure 11.

DPLL[2:0] and the TS_DCO are considered Combo Bus transmitters; the APLL and FOD[2:0] are considered Combo Bus receivers (see Figure 5 and Figure 7). Each Combo Bus receiver generates a frequency control word (FCW) for its synthesizer based on the frequency steering words from the Combo Bus and the setting of the APLL_COMBO_OP_CNFG or FOD_CNFG register fields. Each Combo Bus receiver can be independently configured to add, subtract, or ignore the frequency steering words from each Combo Bus transmitter.

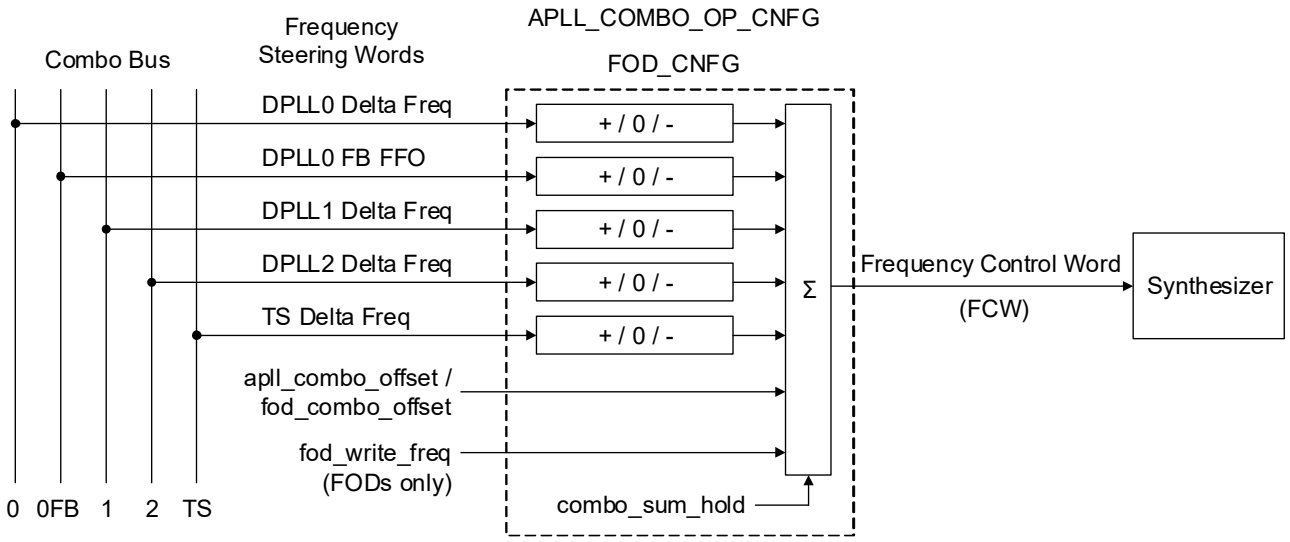


Figure 5. Combo Bus Receiver Block Diagram

1.2.1 Example Combo Bus Receiver Configuration

The Renesas IC Toolbox (RICBox) GUI includes a Combo Bus page that simplifies Combo Bus receiver configuration. The RICBox GUI enforces rules that prevent configuration errors to ensure the device meets application requirements.

Figure 6 illustrates an example application with 3 independent DPLLs and 1 independent DCO. In this application, DPLL0 takes its feedback from and steers the APLL, DPLL1 takes its feedback from and steers FOD1, DPLL2 takes its feedback from and steers FOD2, and the TS_DCO steers FOD0. Table 2 shows the Combo Bus receiver configurations that implement the example application. The configurations are described below.

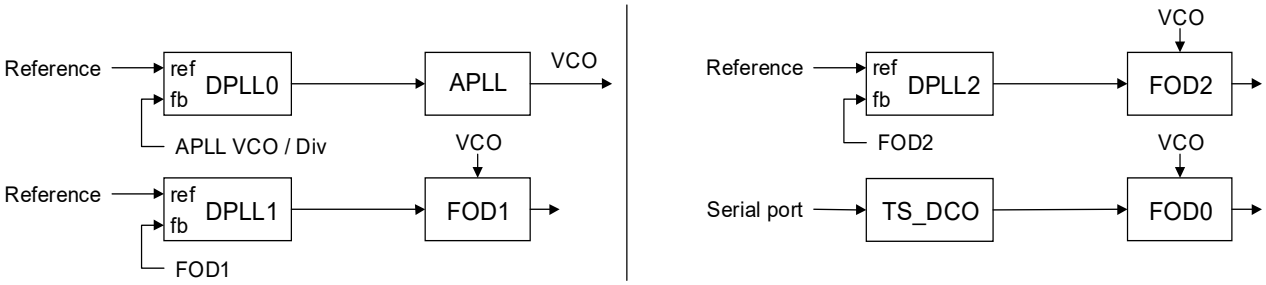


Figure 6. Example 3 DPLL Plus 1 DCO Application

The APLL Combo Bus receiver is configured to calculate its FCW by adding the DPLL0 Delta Freq word to its combo offset while ignoring all other steering signals. In this way the APLL is steered by DPLL0 only. FOD[2:0] derive their output clocks by dividing from the APLL VCO clock, and they will follow digital steering signals applied to the APLL, unless those steering signals are counteracted.

The FOD[2:0] Combo Bus receivers are configured to calculate their FCWs by adding the Delta Freq words from their respective DPLL or DCO to their respective combo offsets and to subtract the DPLL0 Delta Freq word. In this way the FODs counteract the steering signals applied to the APLL and they are steered by their respective DPLLs or DCO only. In this application, the fod_write_freq words are assumed to be held at 0.

The apll_combo_offset, fod_combo_offset, apll_combo_sum_hold, and fod_combo_sum_hold register fields enable external software to suppress frequency transients due to transients on Combo Bus transmitters or reconfigurations of the APLL_COMBO_OP_CNFG or FOD_CNFG register fields.

Table 2. Combo Bus Receiver Configuration for Example 3 DPLL Plus 1 DCO Application

Combo Bus Transmitter	APLL	FOD1	FOD2	FOD0
DPLL0 Delta Freq	Add	Subtract	Subtract	Subtract
DPLL0 FB FFO	Ignore	Ignore	Ignore	Ignore
DPLL1 Delta Freq	Ignore	Add	Ignore	Ignore
DPLL2 Delta Freq	Ignore	Ignore	Add	Ignore
TS_DCO Delta Freq	Ignore	Ignore	Ignore	Add

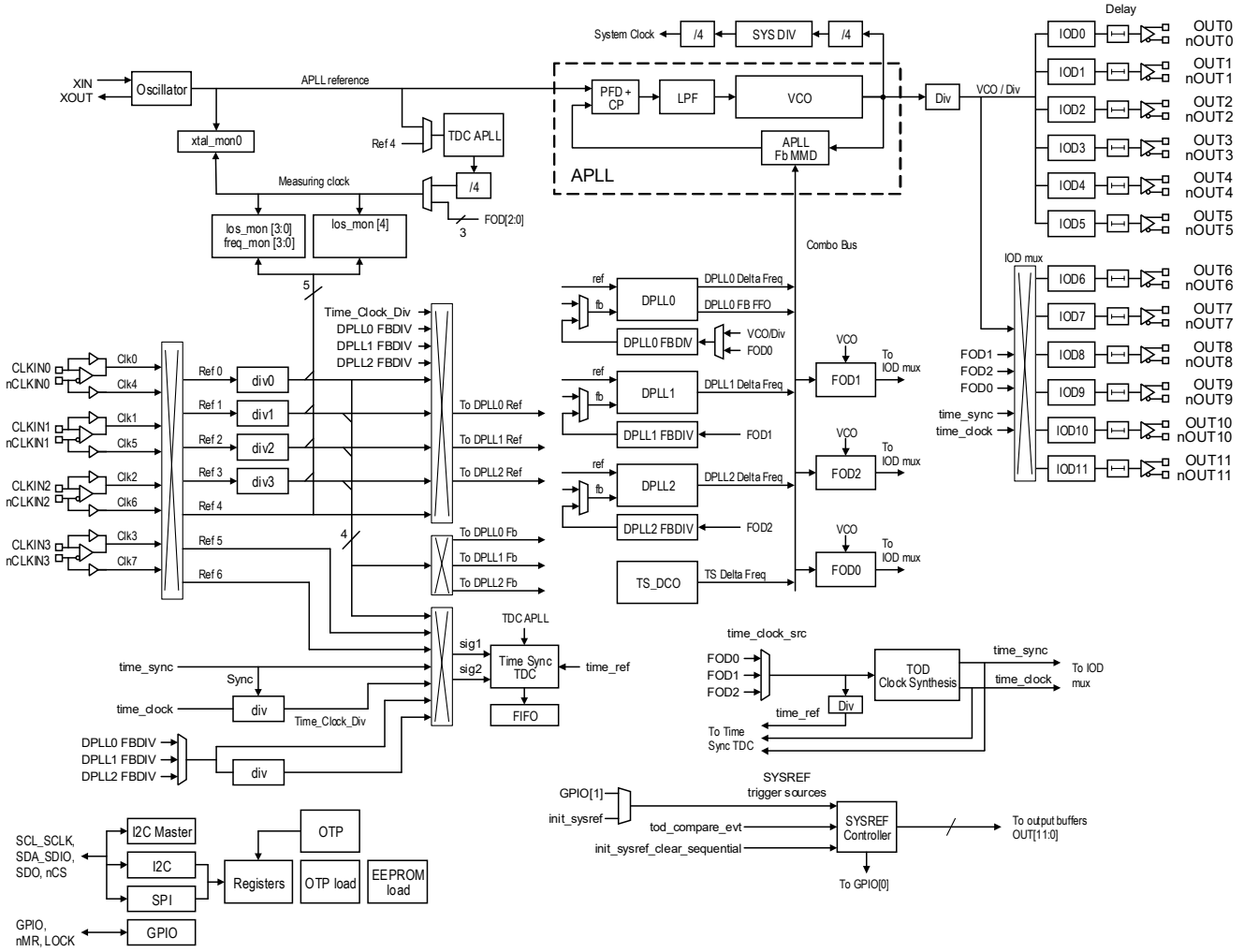


Figure 7. RC38312 Functional Block Diagram

1.3 APLL Frequency Reference

The APLL requires a frequency reference. The frequency reference can be implemented with an external crystal resonator and the device internal oscillator circuitry; or with an external oscillator. If a crystal resonator is used it must be connected between the XIN and XOUT pins. If an external oscillator is used it must be connected to the XIN pin so that it overdrives the device oscillator circuitry. For more information, see the tables titled “Crystal Oscillator Input and APLL AC/DC Electrical Characteristics” and “Recommended Crystal Characteristics” in the applicable device datasheet.

The APLL frequency reference must support the phase noise, frequency accuracy, and frequency stability requirements of the intended application.

For all applications, the phase noise of the frequency reference, after filtering by the APLL, is the minimum phase noise that will appear on all clocks output by the device.

For DPLL applications, the accuracy of the frequency reference determines the frequency accuracy of the reference monitors and free-run clocks. The stability of the frequency reference determines the holdover stability of the DPLLs and it affects the lowest filtering bandwidth the DPLLs can support.

For DCO applications, the accuracy of the frequency reference determines the frequency accuracy of the free-run clocks. The stability of the frequency reference determines the stability of the DCO clocks when a source of synchronization is not available, and it affects the lowest filtering bandwidth that a filtering algorithm can support.

1.4 Analog PLL

The APLL is configured using registers in the APLL section.

The internal APLL locks to the APLL frequency reference and synthesizes an ultra-low phase noise clock of virtually any frequency within its frequency range (see f_{VCO} in the device datasheet). The voltage controlled oscillator (VCO) is the synthesizer for the APLL. The VCO output frequency is equal to the APLL reference frequency multiplied by the APLL divide ratio. The APLL nominal divide ratio is programmed using control registers: apll_fb_div_int and apll_fb_div_frac as indicated in the following equation.

$$\text{APLL divide ratio} = \left(\text{apll\_fb\_div\_int}[9:0] + \frac{\text{apll\_fb\_div\_frac}[37:0]}{2^{38}}\right)$$

The undivided VCO clock is supplied to FOD[2:0] where it is divided as needed (see Fractional Output Dividers).

The VCO clock is divided by vco_div and is available as an input to the integer output dividers (IOD). The divided APLL output clock is available directly to IOD[5:0], and it is available via cross connect to IOD[11:6].

The VCO FFO can be digitally steered by frequency control words (FCW) derived from frequency steering words received from the Combo Bus, as shown in the following equation.

$$\text{APLL feedback divide ratio} = \left(\text{apll\_fb\_div\_int}[9:0] + \frac{\text{apll\_fb\_div\_frac}[37:0]}{2^{38}}\right) x \left(1 + \frac{FCW}{2^{44}}\right)$$

For applications where the APLL is used as an integer-only multiplier, the integer_mode register bit can be set to 0x1 for lowest phase noise. In this mode the apll_fb_div_frac is ignored and the DPLLs and DCOs will be unable to steer the APLL FFO.

1.5 Fractional Output Dividers

The FODs are configured using registers in the FOD section. The FODs synthesize low-phase noise clocks with programmable frequencies by dividing the VCO clock using fractional division. The FOD output clocks are available, via cross connect, to IOD[11:6].

The FODs have the following three modes of operation: FOD Synthesizer Mode, FOD Digitally Controlled Oscillator Mode, and FOD Synchronous Mode. All FOD modes divide the VCO clock, the modes differ in their intended applications and how the divide ratios are specified.

Unused FODs can be powered down to reduce power consumption by setting the respective fod_enable and fod_en_ldo register bits to 0x0.

1.5.1 FOD Synthesizer Mode

In synthesizer mode (fod_sync_mode = 0x0), an FOD divides the VCO frequency by a static value comprised of an integer plus a fraction. In this mode an FOD can translate the VCO frequency to virtually any frequency within the FOD frequency range with 0.9 parts per trillion (PPT) FFO vs the VCO frequency. In synthesizer mode the

FOD divide ratio is programmed using the fod_div_integer, and fod_div_fraction register fields as indicated in the following equation.

$$\text{FOD divide ratio} = \left(\text{fod\_div\_integer}[48:40] + \frac{\text{fod\_div\_fraction}[39:0]}{2^{40}}\right)$$

An FOD in synthesizer mode can counteract digital frequency steering applied to the VCO by a DPLL or a DCO via the Combo Bus. In this way each FOD can be an independent clock domain.

1.5.2 FOD Digitally Controlled Oscillator Mode

In DCO mode (fod_sync_mode = 0x0), an FOD synthesizes a nominal frequency in the same way as the synthesizer mode, and it allows external software to dynamically steer the nominal frequency with 44-bit (0.05 PPT) resolution over a range of ±244 PPM. In DCO mode the nominal FOD divide ratio is programmed using the fod_div_integer, and fod_div_fraction register fields, and the divide ratio is dynamically steered using the fod_write_freq register field, see the following equation. Note that the fod_write_freq register field bypasses the Combo Bus.

$$\text{FOD divide ratio} = \left(\text{fod\_div\_integer}[48:40] + \frac{\text{fod\_div\_fraction}[39:0]}{2^{40}}\right) \times \left(1 + \frac{\text{fod\_write\_freq}[32:0]}{2^{44}}\right)$$

An FOD in FOD DCO mode can counteract digital frequency steering applied to the VCO by a DPLL or a DCO via the Combo Bus. In this way each FOD can be an independent clock domain.

1.5.3 FOD Synchronous Mode

In synchronous mode (fod_sync_mode = 0x1), an FOD divides the VCO frequency by a static value comprised of an integer plus a fraction. The synchronous mode is capable of translating many common VCO frequencies to many common FOD frequencies with zero PPT FFO vs the VCO frequency. The FOD divide ratio in synchronous mode is programmed using the fod_div_integer, fod_div_fraction, fod_div_numerator, and fod_div_denominator register fields as indicated in the following equation.

$$\text{FOD divide ratio} = \left(\text{fod\_div\_integer}[48:40] + \frac{\text{fod\_div\_fraction}[15:0] + \left(\frac{(\text{fod\_div\_numerator}[39:0])}{(\text{fod\_div\_denominator}[39:0])}\right)}{2^{16}}\right)$$

An FOD in synchronous mode will always faithfully follow frequency steering applied to the VCO by a DPLL, or by a DCO or by steering the APLL Frequency Reference (e.g., a VCXO overdriving the XIN pin).

1.6 Divider Synchronization

For DPLL0, the DPLL feedback divider, APLL and associated IODs are automatically synchronized after the device is configured on startup. For DPLL[2:1], the DPLL feedback divider, FOD and IODs are automatically synchronized after the device is configured on startup.

The DPLL[2:0] feedback dividers, FODs and IODs can be manually synchronized by setting divider_sync = 0x1; or by setting apll_reinit = 0x1 (for more information, see Soft Reset Sequence). Re-synchronizing the dividers ensures a deterministic input to output phase relationship for each DPLL.

1.7 DPLL Reference Selection

The DPLLs can lock to any of the clock input references Ref[3:0]. Reference selection can be manual or automatic as determined the dpll_ref_sel_mode register field.

1.7.1 Manual Reference selection

For manual reference selection, the DPLL reference is selected by register or by pin as determined by the dpll_ref_sel_mode register field.

In the case of selection by register, the reference is selected using the dpll_ref_sel register field.

In the case of selection by pin, the reference is selected according to Table 3 using the two pins assigned as DPLL REFIN_SEL[0] and DPLL REFIN_SEL[1] in the gpio_func register field. If DPLL REFIN_SEL[0] or DPLL REFIN_SEL[1] is not assigned to a pin then it is given a value of 0.

Table 3. Reference Selection by Pin

REFIN_SEL[1]	REFIN_SEL[0]	Selected Reference
0	0	Ref 0
0	1	Ref 1
1	0	Ref 2
1	1	Ref 3

1.7.2 Automatic Reference Selection

For automatic reference selection, the reference is selected based on clock quality status and priority. The quality status is provided by the clock monitors. The priorities can be re-programmed in the DPLL_REF_PRIORITY_CNFG register fields. If two clock inputs are programmed to the same priority, the one with the lower index number takes precedence (e.g., Ref0 takes precedence over Ref1).

Automatic reference selection can be revertive or non-revertive as determined by the dpll_revertive_en register field.

In revertive mode, the qualified reference with the highest priority is always selected. If a reference of higher priority than the currently selected reference becomes qualified, the DPLL will switch to that reference. If a reference clock of equal or lower priority than the currently selected one becomes qualified, the DPLL will continue with the current reference.

In non-revertive mode, when a reference with higher priority than the current reference changes status from disqualified to qualified the DPLL will continue with the current reference unless the current reference becomes disqualified. If the current reference becomes disqualified then the DPLL will select the highest priority qualified reference available.

1.7.3 Aligned DPLL Reference Switching

Aligned reference switching is the natural behavior of a DPLL. The DPLL will act to close a phase offset between the selected reference and the feedback clock. Aligned reference switching is enabled by setting dpll_hitless_en = 0x0. A step change in the phase difference can occur when a new reference is selected while the DPLL is in the Acquire State or the Normal State, or when the DPLL exits the Holdover State and enters the Acquire State. The resulting phase transient is filtered by the DPLL loop filter and the phase slope limiter, and there will not be any sudden phase steps or glitches on the device outputs.

Aligned reference switching should be used for applications that require a known phase relationship between the DPLL reference and output clocks.

1.7.4 Hitless DPLL Reference Switching

Hitless reference switching causes the DPLL to ignore the initial phase offset between a newly selected reference and the DPLL feedback clock so that the DPLL can lock to the new reference with a minimal phase transient. Hitless reference switching is enabled by setting dpll_hitless_en = 0x1. Hitless reference switching requires the Hitless Switch State which is available only when the DPLL state machine is in the automatic mode (see Table 4).

A hitless reference switch event begins when the DPLL is in the Holdover State; the event is triggered according to the logic shown in Figure 13. The DPLL measures the phase offset between the selected reference and the DPLL feedback clock. The measured hitless switching phase offset is stored and is subtracted from later phase offsets measured by the DPLL phase detector (for more information, see the Hitless Switch State). The hitless switching phase offset can be cleared by setting hs_offset_clr_b = 0x0.

If there is an FFO between the newly selected reference and the DPLL holdover frequency then a phase transient will occur while the DPLL is in the Acquire State regardless of the hitless reference switching process.

When hitless switching is enabled, the DPLL_PHASE_OFFSET_CNFG register fields are ignored by the DPLL and only the hitless switching phase offset affects the input-output phase offset. When hitless switching is disabled (dpll_hitless_en = 0x0), the hitless switching phase offset is set to zero.

Hitless reference switching does not allow a known phase relationship to exist between the DPLL reference and its output clocks. Hitless reference switching should not be enabled for applications that require a known phase relationship between the DPLL reference and the output clocks.

1.7.5 DPLL External Feedback

In some applications it is useful to use an external feedback path for a DPLL. The dpll_fb_sel register field can be used to select one of the input clock references as the DPLL feedback clock. When external feedback is used, the feedback clock must have the same frequency as the selected DPLL reference.

When external feedback is used, power consumption can be reduced by disabling the DPLL feedback divider. The external feedback dividers are disabled as follows: for DPLL0, by setting vco_dpll_fb_div_en = 0x0 and fod_dpll_fb_div_en = 0x0; for DPLL[2:1], by setting fod_dpll_fb_div_en = 0x0.

1.8 Reference and Crystal Monitors

The reference monitors are configured using the XTALMON, LOSMON, and FREQMON registers.

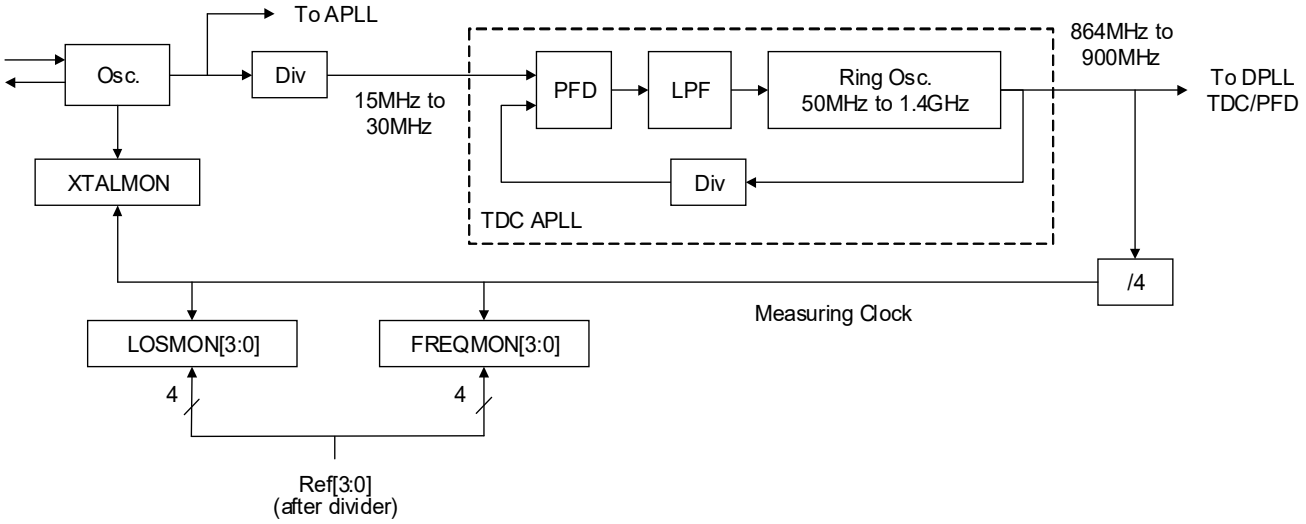


Figure 8. Clock Monitors and TDC APLL

1.9 Digital PLLs

The DPLLs are configured using registers in the DPLL section.

Up to four of the clock inputs can be selected as inputs for the reference monitors and the DPLL reference selection multiplexer using the REF_SEL_CNFG register field. The DPLLs can lock to reference frequencies between 1kHz and 33MHz; clock input frequencies above 33MHz must be divided to 33MHz or lower using the internal reference dividers that are configured using the INPUT_DIV_CNFG register field. The DPLLs steer their synthesizers (APLL or FOD) using digital frequency steering words via the Combo Bus.

1.9.1 DPLL0

The block diagram of DPLL0 is shown in Figure 9. DPLL0 is the only DPLL that allows its feedback clock to be sourced from VCO/Div. DPLL0 can use VCO/Div or FOD0 as its feedback clock source. The feedback source is selected using the dpll_fbdiv_src_clk_sel register bit.

DPLL0 can be configured to operate as a DCO while the DPLL loop is closed (Normal State or Acquire State). In this configuration the DPLL feedback divider can be controlled via the dpll_fb_write_freq register field to allow the output frequency to be steered by external software. This function is typically used with DPLL0 locked to an oven-controlled crystal oscillator or a temperature-compensated crystal oscillator; the result is a stable, ultra-low phase noise DCO.

DPLL0 can be configured to operate as a DCO when it is not locked to a reference (Write Frequency State or Write Phase Mode). In this configuration the DPLL feedback loop is not closed and the associated synthesizer is steered by external software controlling the write_phase or write_freq register fields.

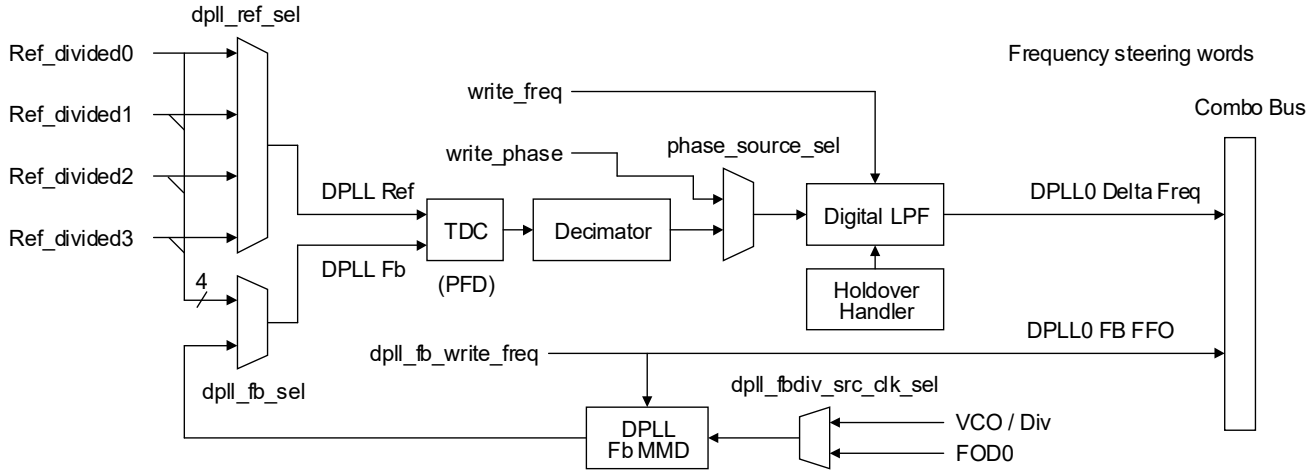


Figure 9. DPLL0 Block Diagram

1.9.2 DPLL1 and DPLL2

The block diagram of DPLL1 and DPLL2 is shown in Figure 10. The feedback signals from DPLL1 and DPLL2 are always sourced from their respective FODs.

DPLL1 and DPLL2 can be configured as DCOs when they are not locked to a reference (Write Frequency State or Write Phase Mode). In this configuration the feedback loops are not closed and the associated synthesizers are steered by external software controlling the write_phase or write_freq register fields.

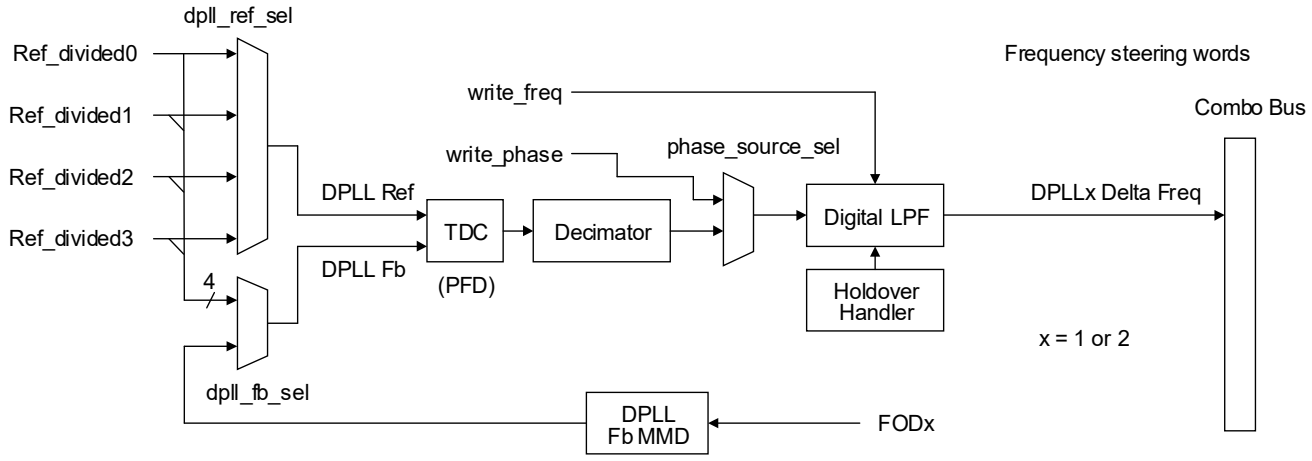


Figure 10. DPLL1 and DPLL2 Block Diagram

1.9.3 Time Sync DCO

The block diagram of the Time Sync DCO (TS_DCO) is shown in Figure 11. The TS_DCO is steered by external software; it does not include a phase frequency detector (PFD) and it does not use a feedback signal. The TS_DCO can be steered using the lpf_mode register field and the lpf_wr_phase or lpf_wr_freq register fields.

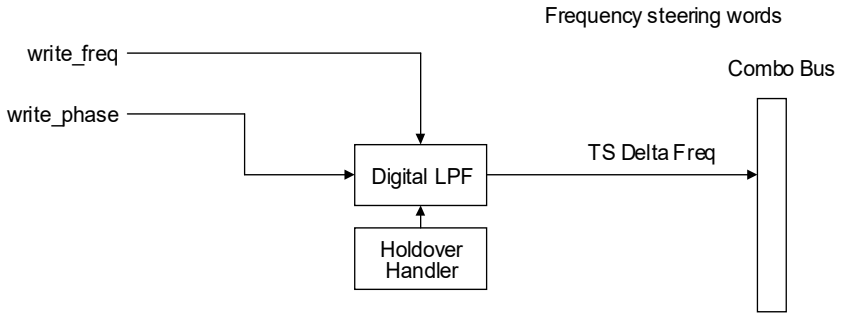


Figure 11. TS_DCO Block Diagram

The frequency steering words output by a DPLL or a DCO are the sum of the proportional term and the integral term output by the DPLL digital low-pass filter and the xtal_trim register field (see Figure 12). The integral term represents a time average of the frequency steering words. The proportional term represents the short-term changes of the frequency steering words due to tracking jitter and phase transients on the selected clock input reference.

The DPLL implements a programmable limiter for the magnitude of the proportional term (see phase_slope_limit). This limiter can be used to control the rate of phase change when the DPLL switches between two references with the same FFO. The DPLL also implements a limiter for the magnitude of the integral term (see integrator_limit). This limiter can be used to prevent the holdover value from being pulled outside specified limits during a locking transient. Both limiters are shown in Figure 12.

The xtal_trim register field allows host software to apply an FFO to the APLL frequency reference. The xtal_trim field can be used to improve the free-run frequency accuracy of DPLL/DCO by compensating for a known FFO of the APLL frequency reference. Note that the reference monitors are unaffected by compensation using xtal_trim.

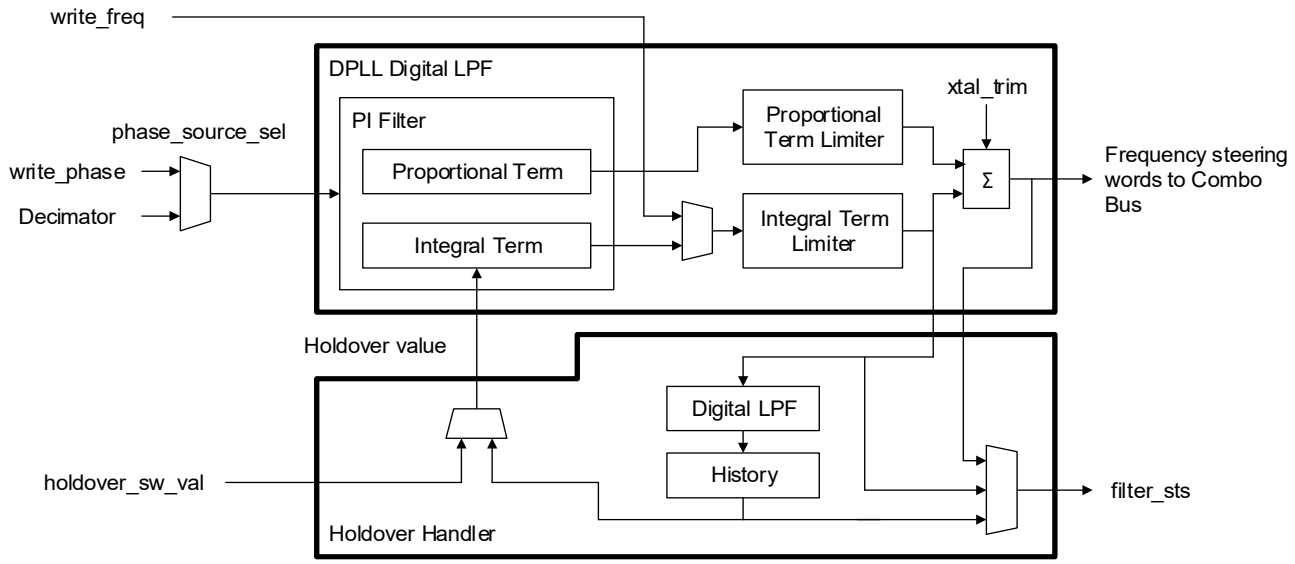


Figure 12. DPLL Digital Low-pass Filter, Limiters, and Holdover Handler

1.9.4 Holdover Handler

The Holdover Handler processes the DPLL integral term and determines the holdover value that is restored to the DPLL integrator when the DPLL enters the Holdover State (see Figure 12).

The Holdover Handler filters the DPLL integrator term with a bypass-able digital low-pass filter. The bandwidth of the holdover filter is determined by the holdover_bw_shift and holdover_bw_mult register fields. The holdover filter does not affect the DPLL transfer function.

The holdover filter can be cleared by writing 0x1 to the holdover_filter_rst bit; this will cause it to be loaded with the current integrator term. Writing 0x0 to the holdover_filter_rst bit will cause the holdover filter to resume filtering.

The Holdover Handler stores the filtered value in two holdover history registers, alternating between them. The update interval for the holdover history registers is defined by the holdover_history register field. When the DPLL enters the Holdover State, the oldest history register value is selected to be restored to the DPLL integrator.

The holdover history can be cleared by writing 0x1 to the holdover_filter_rst bit. This can only be done when the DPLL is not already in the Holdover State.

Host software can override the holdover value processed by the Holdover Handler by writing a holdover value in the holdover_sw_val register field and setting the manual_holdover bit.

The Holdover Handler makes the following values available to host software via the filter_sts register field: the current integral term, the current sum of the proportional and integral terms, and the oldest holdover history value. The value selected is determined by the filter_status_sel register field.

1.9.5 DPLL State Machine

The DPLL can operate in any of six states: Free-run State, Normal State, Holdover State, Write Frequency State, Acquire State, and Hitless Switch State; while in the Normal state or the Acquire state it can also operate in the Write Phase Mode. The state or mode of the DPLL is controlled by the dpll_en register bit and the dpll_mode register field as shown in Table 4. The Hitless Switch State is part of the automatic state machine; the DPLL cannot be specifically forced into this state by host software.

The current DPLL state is indicated by the dpll_state_sts register field. DPLL state changes can be monitored using the dpll_state_ch_int_sts interrupt status bit.

Table 4. DPLL States and Modes

Manual State Machine or Automatic State Machine	Forced State or Mode	dpll_en	dpll_mode
Manual See Figure 14	Freerun	0x0	0x0, 0x6
	Reserved		0x2, 0x3, 0x4, 0x5, 0x7
	Reserved	0x1	0x5, 0x7
	Freerun		0x0
	Normal		0x1
	Holdover		0x2
	Write Frequency		0x3
	Acquire		0x4
Automatic See Figure 13	Automatic Mode		0x6

1.9.6 Free-run State

In the Free-run state, the integral and proportional terms output by the DPLL are held at zero. In this state, the FFO of the APLL is determined by the FFO of the APLL frequency reference, plus xtal_trim if used. During the Master Reset Sequence or the Soft Reset Sequence the DPLL will be in the Free-run state (see Figure 13 and Figure 14). There are three combinations of the dpll_en register bit and the dpll_mode register field that will force the DPLL into the Free-run state (see Table 4).

The DPLL can automatically enter the Free-run state; the conditions for entering this state depend on the operating mode of the DPLL state machine. When the DPLL state machine is in the manual mode it will automatically enter the Free-run state according to the logic shown in Figure 14. When the DPLL state machine is in the automatic mode, the DPLL will automatically enter the Free-run state according to the logic shown in Figure 13. See Table 4 to configure the operating mode of the DPLL state machine.

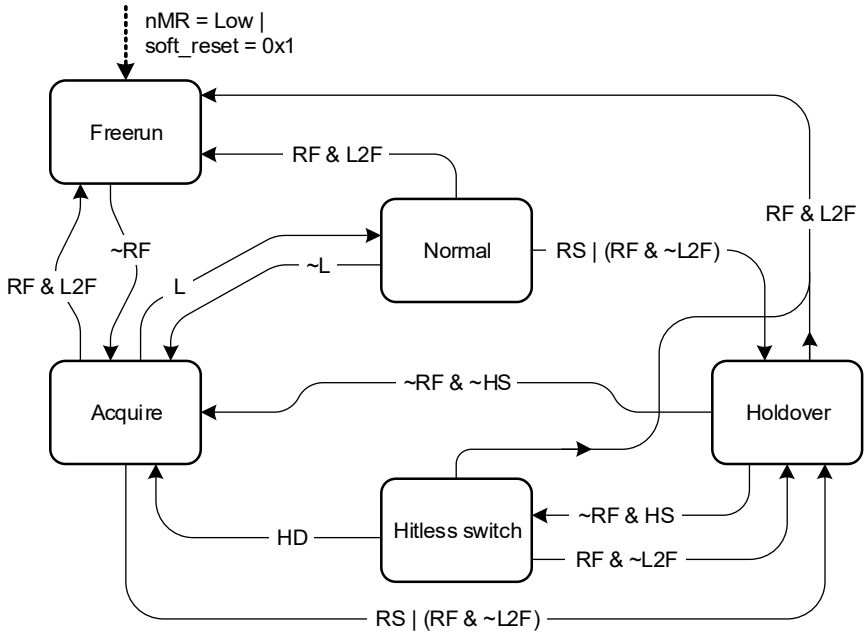
1.9.7 Normal State

The Normal state supports DPLL operation, the DPLL control loop is active and it steers the assigned synthesizer (APLL or FOD). While the DPLL is locked, OUTx clocks sourced from the DPLL synthesizer track the selected reference according to the configured DPLL bandwidth, damping factor, FFO limit, and phase slope limit. The DPLL is placed in the Normal state by the following settings: dpll_en = 0x1 and dpll_mode = 0x1 (see Table 4).

When the DPLL state machine is in the automatic mode, the DPLL will automatically enter the Normal state according to the logic shown in Figure 13. See Table 4 to configure the operating mode of the DPLL state machine.

In the Normal state the DPLL bandwidth and damping are configured using the normal_bw_shift and normal_bw_mult; and normal_damping_shift and normal_damping_mult register fields respectively. The FFO limit and the phase slope limit are configured using the integrator_limit and phase_slope_limit register fields, respectively.

The Normal state is intended for normal operation when the DPLL is locked and tracking its reference. The DPLL bandwidth, damping, FFO limit and phase slope limit should be configured to meet the standard requirements of the application.



Acronyms

RF	Reference Fail	Internal signal
L	Lock	Internal signal
L2F	LOS to Freerun	los_to_freerun bit
HS	Hitless Switch	dpll_hitless_en bit
HD	Hitless Switch Done	Internal signal
RS	Reference Switch	Internal signal

Logic Symbols

"&"	AND
"|"	OR
"~"	NOT

Line Types

Continuous lines indicate automatic state transitions
Dotted lines indicate register or pin status

Figure 13. DPLL State Machine in Automatic Mode

1.9.8 Write Phase Mode

The Write Phase mode supports DCO operation when the DPLL loop is not closed. It allows host software to steer the FFO of the assigned synthesizer (APLL or FOD) using the write_phase register field. The Write Phase mode is enabled when the DPLL is in the Normal State or the Acquire State and phase_source_sel = 0x1.

In the Write Phase mode the DPLL loop is open and the output of the PFD, via the decimator, is replaced by values from the write_phase register field (see Figure 12).

1.9.9 Holdover State

The Holdover state supports continued generation of accurate clock frequencies during short-term interruptions of the synchronization reference. The DPLL can be forced into the Holdover state by the following settings: dpll_en = 0x1 and dpll_mode = 0x2 (see Table 4).

The DPLL can automatically enter the Holdover state. The conditions for entering this state depend on the operating mode of the DPLL state machine. When the DPLL state machine is in the manual mode it will automatically enter the Holdover state according to the logic shown in Figure 14. When the DPLL state machine is in the automatic mode, the DPLL will automatically enter the Holdover state according to the logic shown in Figure 13. See Table 4 to configure the operating mode of the DPLL state machine.

In the Holdover state the integral term from the low-pass filter output is replaced by a holdover value from holdover handler and the proportional term is held at zero, see Figure 12.

1.9.10 Write Frequency State

The Write Frequency state supports DCO operation when the DPLL loop is not closed. It allows host software to steer the FFO of the assigned synthesizer (APLL or FOD) using the write_freq register field. The DPLLDCO can be forced into the Write Frequency state by the following settings: dpll_en = 0x1 and dpll_mode = 0x3 (see Table 4).

In the Write Frequency state the integral term from the low-pass filter output is replaced by values from the write_freq register field and the proportional term is held at zero (see Figure 12).

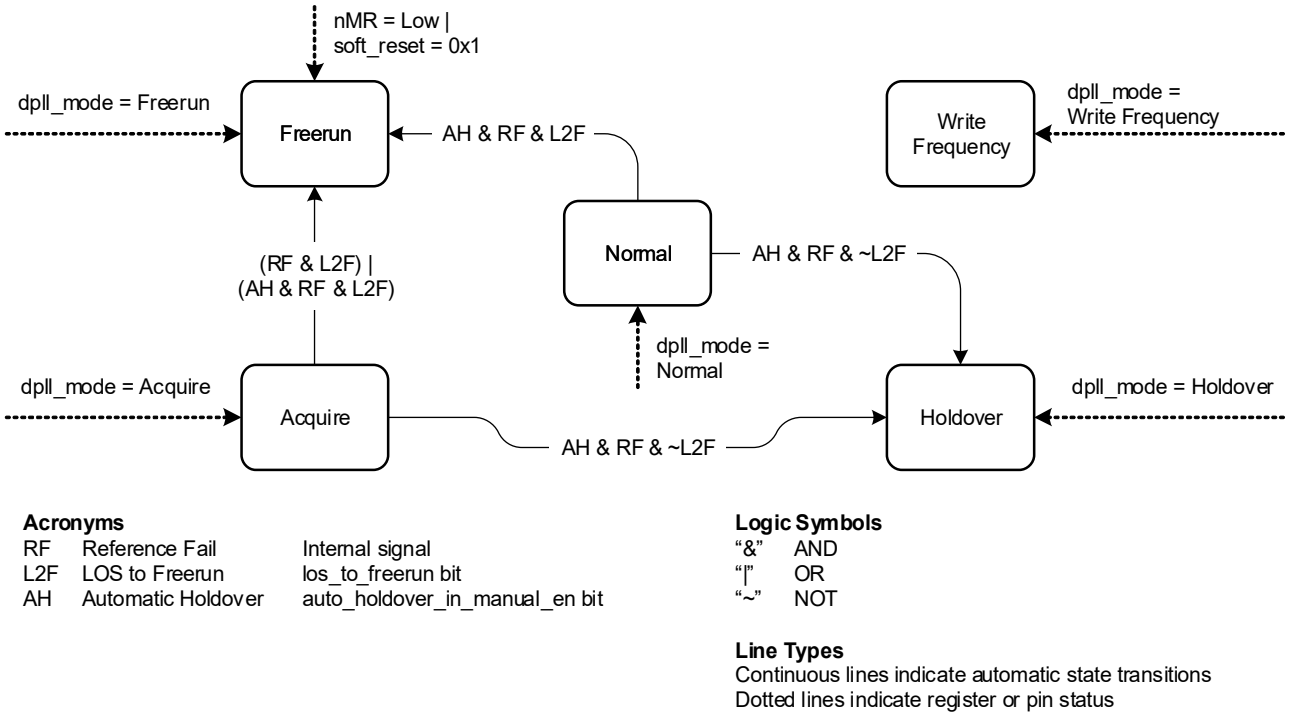


Figure 14. DPLL State Machine in Manual Mode

1.9.11 Acquire State

The Acquire state supports DPLL operation, the DPLL control loop is active and it steers the APLL. The Acquire state is the same as the Normal State except that it supports independent DPLL bandwidth and damping settings and it supports different automatic state transitions (see Figure 13 and Figure 14). The DPLL can be forced into the Acquire state by the following settings: dpll_en = 0x1and dpll_mode = 0x4 (see Table 4).

When the DPLL state machine is in the automatic mode, the DPLL will automatically enter the Acquire state according to the logic shown in Figure 13. See Table 4 to configure the operating mode of the DPLL state machine.

In the Acquire state the DPLL bandwidth and damping are configured using the acquire_bw_shift and acquire_bw_mult; and acquire_damping_shift and acquire_damping_mult register fields respectively. The FFO limit and the phase slope limit are configured using the integrator_limit and phase_slope_limit register fields respectively.

The Acquire state is intended to accelerate the DPLL locking process with relaxed DPLL bandwidth and damping versus the Normal state. To further accelerate the locking process, host software can also relax the phase slope limit while the DPLL is in the acquire state by configuring the phase_slope_limit register field. After the DPLL reports lock, host software can restore the normal phase slope limit.

1.9.12 Hitless Switch State

The Hitless Switch state manages the process of switching the DPLL from the Holdover State to the Acquire State without causing a phase transient on the OUTx clocks (see Hitless DPLL Reference Switching). The Hitless state is accessible when the DPLL is in the Holdover State and the DPLL state machine is in the automatic mode. The Hitless state cannot be directly accessed using the dpll_mode register field.

When the DPLL state machine is in the automatic mode, the DPLL will automatically enter the Hitless Switch state according to the logic shown in Figure 13. See Table 4 to configure the operating mode of the DPLL state machine.

The hitless switching phase offset is measured over the number of reference clock cycles programmed in the hs_counter_limit register field. The phase offset measurements are averaged by the decimator using the bandwidth for hitless switching defined by the dec_hitless_bw_shift register field. When the phase measurement is complete the DPLL enters the Acquire State (see Figure 13).

For applications where the DPLL state machine is used in the manual mode the Hitless Switch state is accessible as follows: after the DPLL has entered the Holdover State according to the logic in Figure 14, set the DPLL state machine to automatic mode by setting dpll_mode = 0x6. When the selected reference is valid the DPLL will transition from the Holdover State to the Hitless Switch state and then to the Acquire State (see Figure 13). Host software can monitor the automatic state machine using the dpll_state_sts register field and the dpll_state_ch_int_sts interrupt status bit and return to the manual mode as desired.

1.10 Clock Output Paths

The device has two types of clock output path. The paths for OUT[5:0] have access to the VCO/Div clock only. The paths for OUT[11:6] have access to the VCO/Div clock and the FOD[2:0] clocks. Each clock output path includes one IOD and one clock output buffer.

1.10.1 Clock Output Enable

The device enables and disables clock outputs synchronously to ensure there are no runt pulses during clock output enable and disable operations.

When global_oe = 0x0, all clock outputs are disabled. When global_oe = 0x1, clock outputs with oe_source_sel = 0x1 can be enabled or disabled individually using the respective out_driver_en register bit. When global_oe = 0x1, clock outputs with oe_source_sel = 0x0 can be enabled or disabled in groups or individually under GPIO control. See Table 5 for a summary.

Table 5. Clock Output Enable Control Truth Table

global_oe	oe_source_sel[1]	out_driver_en[2]	GPIO Input Logic Level[3][4]	Clock Output
0x0	X	X	X	Disabled
0x1	0x0	X	Low	Disabled
		X	High	Enabled[5]
	0x1	0x0	X	Disabled
		0x1	X	Enabled[5]

1. There is one oe_source_sel register bit per clock output.

2. There is one out_driver_en register bit per clock output.
3. Assuming gpio_pol = 0x0 for the GPIO, otherwise the GPIO logic levels are inverted.
4. Global OE, Group OE or OE as assigned using the gpio_func register field, see Table 6.
5. During startup, the clock output behaves according to the out_startup register field.

GPIOs can be assigned one of the following clock output enable functions using the gpio_func register field: Global OE, Group OE, or OE. When a GPIO is assigned the Global OE function (gpio_func = 0x4C), that pin controls the output enable for all clock outputs with oe_source_sel = 0x0. When a GPIO is assigned the Group OE function (gpio_func = 0x4D to 0x4F) and no GPIO is assigned the Global OE function, that pin controls the output enable for all clock outputs in the group with oe_source_sel = 0x0. When a GPIO is assigned the OE function (gpio_func = 0x50 to 0x5B) and no GPIO is assigned the Global OE function or the Group OE function, that pin controls the output enable for its assigned clock output if the clock output has oe_source_sel = 0x0. See Table 6 for a summary.

Table 6. GPIO Clock Output Enable Control Hierarchy and Truth Table[1]

Global OE[2] GPIO Input Logic Level	Group OE[3] GPIO Input Logic Level	OE[4] GPIO Input Logic Level	Clock Output
Low	X	X	Disabled
High	X	X	Enabled[5]
No GPIO is assigned the Global OE function	Low	X	Disabled
	High	X	Enabled[5]
	No GPIO is assigned the Group OE function	Low	Disabled
		High	Enabled[5]
		No GPIO is assigned the OE function	

1. Assuming gpio_pol = 0x0 for the GPIO, otherwise the GPIO logic levels are inverted.
2. gpio_func = 0x4C, applies to all clock output pins with oe_source_sel = 0x0.
3. gpio_func = 0x4D to 0x4F, applies to clock output pins in the group with oe_source_sel = 0x0.
4. gpio_func = 0x50 to 0x5B, applies to the clock output pin if the respective oe_source_sel = 0x0.
5. During startup, the clock output behaves according to the out_startup register field.

When more than one GPIO is assigned the same output enable function, the GPIO with the lower index takes precedence (for example, GPIO[0] takes precedence over GPIO[1]).

When a GPIO is assigned an output enable function it should be configured with gpio_resync = 0x0 and gpio_deglitch_bypass = 0x1; this disables GPIO resynchronization and bypasses the GPIO deglitcher ensuring lowest latency. When configured this way, the maximum time delay from the time the voltage level on a GPIO input changes state until the clock output is enabled or disabled is 13ns plus four periods of the output clock.

1.10.2 Integer Output Dividers

The IODs are configured using registers in the IOD section.

The 23-bit integer divide ratio for each IOD is programmed using the respective iod_divider register field. Programming an IOD with a value of 0x0 or 0x1 causes the divider to be bypassed. When reprogramming an IOD divider value after startup, writing the entire iod_divider value as a single burst will ensure the divider is updated atomically; this prevents unintended intermediate divider values from being latched by the divider. After reprogramming an IOD, a divider synchronization is recommended as described below.

The IODs, FODs, and DPLL feedback divider can be manually synchronized by setting the divider_sync register bit to 0x1, or by setting the apll_reinit register bit to 0x1 (for more information, see Soft Reset Sequence). IOD output clocks will be interrupted during divider synchronization, but will resume after synchronization and no runt pulses will be generated.

1.10.2.1 IOD Phase Adjustment

The phase of each IOD output clock can be independently adjusted using the signed 16-bit iod_phase_config register field. IOD phase adjustments are specified in periods of the IOD input clock.

IOD phase adjustments are applied by temporarily modulating the high phase of the IOD output clock. Positive phase adjustments will extend the duration of the high phase and negative phase adjustments will reduce the duration of the high phase. The entire phase adjustment in the iod_phase_config register field will be applied regardless of whether it spans multiple cycles of the IOD output clock. The IOD will apply the phase adjustment in a single step or multiple steps, depending on the divide ratio in the iod_divider register field. In some cases, the IOD will apply a phase adjustment in multiple steps of one period of the IOD input clock.

IOD phase adjustments are triggered by writing the iod_phase_config trigger bit to 0x1. The iod_ph_adj_now bit will remain high until the phase adjustment is completed then it will be automatically cleared. An IOD can be configured to automatically apply an IOD phase adjustment after a divider synchronization event by setting iod_ph_adj_post_sync = 0x1.

The pulse width high during phase adjustments will not be less than two periods of the IOD input clock. Negative phase adjustments are not possible if iod_divider ≤ 0x5. IOD phase adjustments are not possible if the IOD is bypassed (e.g., iod_divider = 0x0 or 0x1).

1.10.3 Clock Output Path for OUT[5:0]

The clock source selection for OUT[5:0] is illustrated in Figure 15.

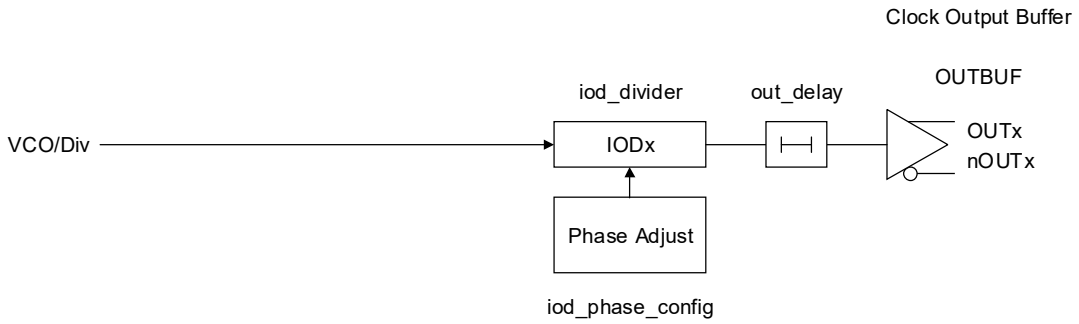


Figure 15. Clock Path for OUT[5:0]

1.10.4 Output Clock Path for OUT[11:6]

The clock source selection for OUT[11:6] is illustrated in Figure 16, the clock source for the output is selected using the iod_mux_sel register field.

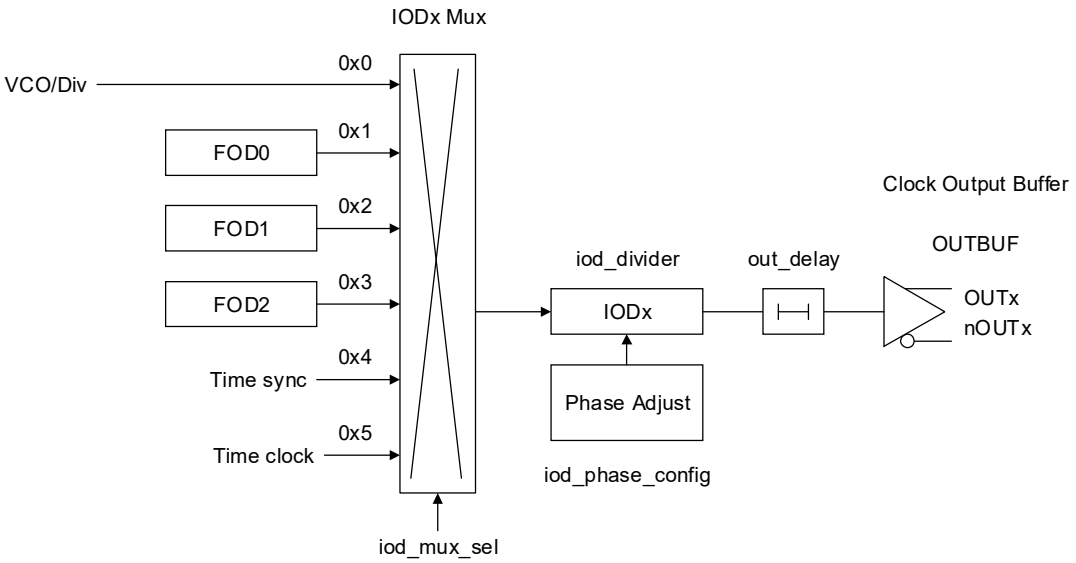


Figure 16. Clock Path for OUT[11:6]

1.10.5 Clock Output Buffers

The clock output buffers are configured using registers in the OUTBUF section. Unused clock paths can be powered down by setting the respective out_pd register bit to 0x1.

1.11 SYSREF

The RC38312, RC38112, RC38208, RC38108 can generate SYSREF and device clock pairs that support JEDEC JESD204B/C/D. SYSREF identifies specific device clock edges for logic devices, analog-to-digital converters, and digital-to-analog converters. See Figure 17 for an illustration of a SYSREF and device clock pair. The phase relationship between SYSREF received at each device in a system must be deterministic.

Any OUT[11:0] can be configured to output SYSREF by setting the respective out_sysref_sel register bit to 0x1.

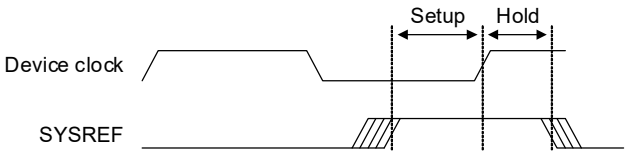


Figure 17. SYSREF and Device Clock Pair at the Receiving Device

1.11.1 SYSREF Controller

SYSREF can be continuous or intermittent as managed by the SYSREF controller. A SYSREF event is defined as a pattern of one or more SYSREF pulses or SYSREF pulses and pauses managed by the SYSREF controller. The SYSREF controller implements the three types of SYSREF events shown in Table 7. During a SYSREF event, the device outputs complete SYSREF cycles with 50/50 duty cycle; the device will not output a runt pulse.

The RC38312, RC38112, RC38208, RC38108 has a single SYSREF controller that manages all SYSREF outputs. The SYSREF controller operates in any of the eight SYSREF modes shown in Table 8. The SYSREF mode is determined by the settings of the sysref_srg and sysref_sro register fields. The same SYSREF mode applies to all SYSREF outputs.

Table 7. SYSREF Event Types

SYSREF Event Type	Description
Continuous	When a start-trigger is detected, the device outputs SYSREF pulses until a stop-trigger is detected.
Counted	When a start-trigger is detected, the device outputs SYSREF pulses until the number of SYSREF pulses given by sysref_pulse_count is reached, then it stops outputting SYSREF pulses.
Continuous pulse/pause	When a start-trigger is detected, the device outputs a SYSREF pattern consisting of SYSREF pulses followed by pauses. The number of pulses is given by sysref_pulse_count, the number of pauses is given by sysref_pause_count. The pattern repeats until a stop-trigger is detected.

Table 8. SYSREF Modes

sysref_srg	sysref_sro [1:0]	SYSREF Mode	Description
0	00	0	Counted SYSREF. The start-trigger is a low-to-high for init_sysref and the stop-trigger is not applicable.
0	01	1	Continuous SYSREF pulse/pause. The start-trigger is a low-to-high for init_sysref and the stop-trigger is a high-to-low for init_sysref.
0	10	2	Continuous SYSREF. The start-trigger is a low-to-high for init_sysref and the stop-trigger is a high-to-low for init_sysref.
0	11	3	Continuous SYSREF. The start-trigger is a rising edge on GPIO[1] and the stop-trigger is a low-to-high for sysref_clear_sequential.
1	00	4	Counted SYSREF. When sysref_tod_trigger_sel = 0x0, the start-trigger is a rising edge on GPIO[1] and the stop trigger is not applicable. When sysref_tod_trigger_sel = 0x1, the start-trigger is a low-to-high transition on tod_compare_evt and the stop-trigger is not applicable.
1	01	5	Counted SYSREF. When sysref_tod_trigger_sel = 0x0, the start-trigger is a falling edge on GPIO[1]. The stop-trigger is not applicable. This mode is not supported when sysref_tod_trigger_sel = 0x1.
1	10	6	Continuous SYSREF. When sysref_tod_trigger_sel = 0x0, the start-trigger is a rising-edge on GPIO[1] and the stop-trigger is a falling-edge on GPIO[1]. When sysref_tod_trigger_sel = 0x1, the start-trigger is a low-to-high transition on tod_compare_evt and the stop-trigger is a high-to-low on tod_compare_evt.
1	11	7	Continuous SYSREF. The start-trigger is a falling-edge on GPIO[1] and the stop-trigger is a rising-edge on GPIO[1]. This mode is not supported when sysref_tod_trigger_sel = 0x1.

1.11.2 SYSREF Trigger Sources

As described in Table 8, each SYSREF mode has a start-trigger and some SYREF modes have a stop-trigger. The trigger sources and the trigger criteria are determined by the SYSREF mode, and for SYSREF modes 4 to 7, by the sysref_tod_trigger_sel register bit.

Figure 18 shows the supported SYSREF trigger sources. Depending on the SYSREF Mode, the SYSREF trigger source can be a register write applied to the init_sysref register bit or the sysref_clear_sequential register bit; or the sysref_tod_trigger_sel register bit can select between an external signal applied to GPIO[1], or the tod_compare_evt register bit.

1.11.3 SYSREF Outputs

Any of OUT[11:0] can be configured as a SYSREF output by writing 0x1 to the respective bit of the out_sysref_sel register field. The SYSREF controller holds the SYSREF output drivers in a disabled state except during SYSREF events. While the SYSREF output drivers are disabled, the SYSREF outputs behave according to the description of the out_driver_en register bit. During SYSREF events, the SYSREF controller enables the SYSREF output drivers for complete cycles of the respective SYSREF_CLK.

1.11.4 SYSREF Phase Alignment

The DPLL0 and APLL combination can be used to generate SYSREF and device clock pairs with excellent phase noise. Alternatively, any of the DPLL and FOD combinations can be used for SYSREF and device clock pairs, the phase noise will be higher than the DPLL0 and APLL combination.

For DPLL channels using internal feedback, the DPLL feedback divider and all IODs connected to the DPLL are synchronized. For DPLL channels using external feedback, the DPLL feedback divider is not used and all IODs connected to the DPLL are synchronized.

Figure 18 illustrates a DPLL channel configured to output SYSREF and device clock pairs. DPLL_REF is the reference for the DPLL; it is traceable to a clock input (CLKIN). DPLL_Fb is the feedback clock; its frequency must equal the frequency of DPLL_REF. When the DPLL is locked and settled, the rising edges of DPLL_Fb are aligned with the rising edges of DPLL_REF. The internal SYSREF_CLKs drive the SYSREF clock outputs. The internal DEV_CLKs drive the device clock outputs.

When using internal feedback, by default, after the DPLL feedback divider and the IODs are synchronized, the first rising edges of DPLL_Fb, the SYSREF_CLKs and DEV_CLKs will be coincident, as illustrated in Figure 19. In addition, the rising edges of these clocks will be coincident at the end of every N^{th} clock cycle, where N for each clock is the frequency of that clock divided by the greatest common divisor of the three clock frequencies. Figure 19 illustrates an example where the frequency ratio of DPLL_Fb to SYSREF_CLK to DEV_CLK is 1:2:8.

The phase of each SYSREF_CLK with respect to the other clocks can be independently adjusted using the respective iod_phase_config and out_delay register fields with the special restriction that the magnitude of the total phase adjustment for any SYSREF_CLK must be less than or equal to one eighth the period of the internal SYSREF_CLK.

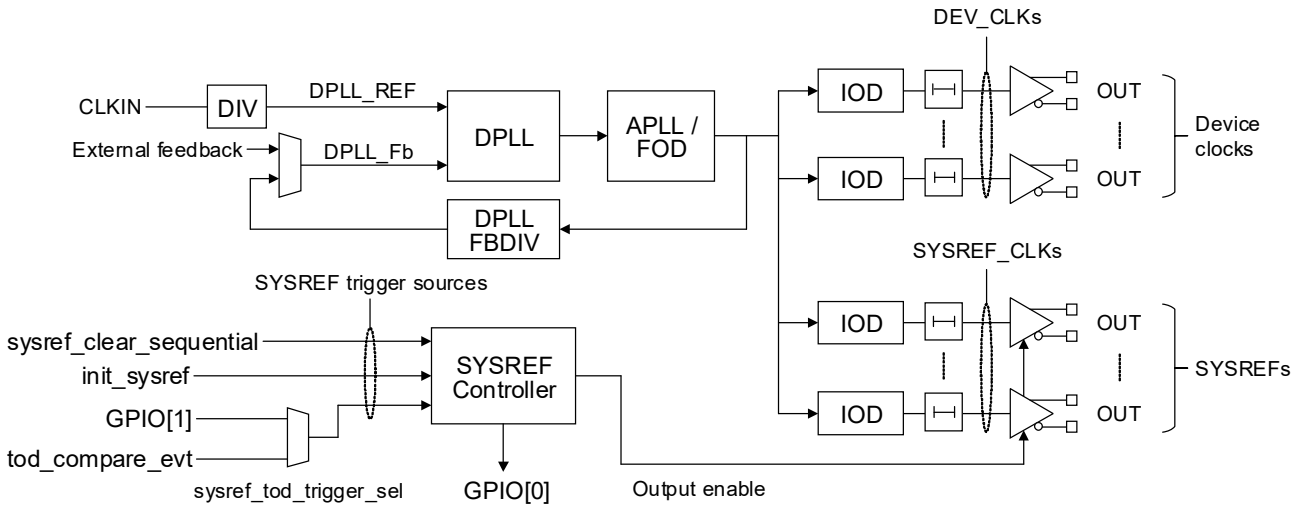


Figure 18. SYSREF and Device Clock Control and Generation Block Diagram

The phase of each DEV_CLK can be independently adjusted with respect to the other clocks using the respective iod_phase_config and out_delay register fields, with no special restriction on the total phase adjustment.

The phases of all SYSREF_CLK and DEV_CLK can be adjusted together with respect to DPLL_REF using the respective dpll_phase_offset register field, with no special restriction on the total phase adjustment.

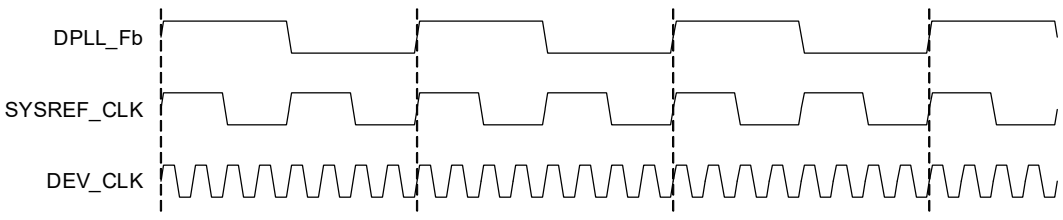


Figure 19. Example of DPLL Reference, SYSREF, and Device Clock Phase Alignment

1.11.5 Single SYSREF Device

The RC38312, RC38112, RC38208, RC38108 supports SYSREF for single devices and for SYSREF groups containing multiple devices. For applications where a single device is used, the recommended settings for the SYSREF group related control registers and GPIO control registers are shown in Table 9.

Table 9. Recommended Configurations for a Single SYSREF Device [1]

SYSREF Mode	Recommended Configuration	Comments
Any	sysref_is_prime = 0x0	The device will not output SYSREF start-triggers and stop-triggers on GPIO[0].
	sysref_trig_from_prime = 0x0	The device will begin the SYSREF event on the second rising edge of SYSREF_CLK after a SYSREF start-trigger is detected. For a quicker response, set sysref_trig_from_prime = 0x1, with this setting the SYSREF event will begin on the first rising edge of SYSREF_CLK after a SYSREF start-trigger is detected.
4 or 6	gpio_func = 0x21 for GPIO[1] pad_gpio_oe_b = 0x1 for GPIO[1]	GPIO[1] will accept SYSREF start-triggers and stop-triggers.

1. See Table 10 for the configuration of devices in a SYSREF group.

1.11.6 SYSREF Groups

The RC38312, RC38112, RC38208, RC38108 supports SYSREF groups. A SYSREF group includes a primary SYSREF device and one or more secondary SYSREF devices, as illustrated in Figure 20 and Figure 25. The primary device accepts asynchronous SYSREF start-triggers and stop-triggers, or SYSREF requests via a register write or GPIO input. The primary device generates synchronous SYSREF start-triggers for itself and for the secondary devices in the group.

The primary and secondary devices in a SYSREF group will react to SYSREF start-triggers according to the setting of the sysref_trig_from_prime register bit so that the primary and secondary device can start their SYSREF events simultaneously. See Table 10 and Table 11 for the recommended configurations of devices in a SYSREF group.

1.11.6.1 SYSREF Group Using SYSREF Mode 0, 1, 2, 4, 5, 6, or 7

Figure 20 shows a SYSREF group using a primary device configured for SYSREF Mode 0, 1, 2, 4, 5, 6, or 7 and secondary devices configured for SYSREF Mode 4 or 6. The primary device accepts asynchronous SYSREF start-triggers and stop-triggers via a register write or GPIO[1] input. The start-triggers and stop-triggers are according to the SYSREF Mode for which the primary is configured.

The primary device outputs synchronous SYSREF start-triggers on GPIO[0] for the secondary devices in the group. The SYSREF start-triggers and stop-triggers output by the primary device on GPIO[0] are compatible with secondary devices using SYSREF Mode 4 for counted SYSREF modes, or SYSREF Mode 6 for continuous SYSREF modes.

Table 10. Recommended Configurations for Devices in a SYSREF Group Using SYSREF Mode 0-2, 4-7

Position in the Group	SYSREF Mode	Recommended Configuration	Comments
Primary	0, 1, 2, 4, 5, 6, 7	sysref_is_prime = 0x1 gpio_func = 0x22 for GPIO[0] pad_gpio_oe_b = 0x1 for GPIO[0]	The primary will output SYSREF start-triggers and stop-triggers, for the secondaries, on GPIO[0]. For more information, see the description of sysref_is_prime.
		sysref_trig_from_prime = 0x0	The primary will begin the SYSREF event on the second rising edge of SYSREF_CLK after a SYSREF start-trigger is detected. The one SYSREF period delay allows the primary to begin the SYSREF event simultaneously with the secondaries.
	4, 6	SYSREF Mode 4 or 6: gpio_func = 0x21 for GPIO[1] pad_gpio_oe_b = 0x1 for GPIO[1]	GPIO[1] will accept SYSREF start-triggers and stop-triggers.

Table 10. Recommended Configurations for Devices in a SYSREF Group Using SYSREF Mode 0-2, 4-7

Position in the Group	SYSREF Mode	Recommended Configuration	Comments
Secondary	4, 6[1]	sysref_is_prime = 0x0	The secondary will not output SYSREF start-triggers and stop-triggers on GPIO[0].
		sysref_trig_from_prime = 0x1	The secondary will begin the SYSREF event on the first rising edge of SYSREF_CLK after a SYSREF start-trigger is detected.
		gpio_func = 0x21 for GPIO[1] pad_gpio_oe_b = 0x1 for GPIO[1]	GPIO[1] will accept SYSREF start-triggers and stop-triggers.

1. Only SYSREF Mode 4 and 6 are compatible with the start-triggers and stop-triggers from the primary GPIO[0].

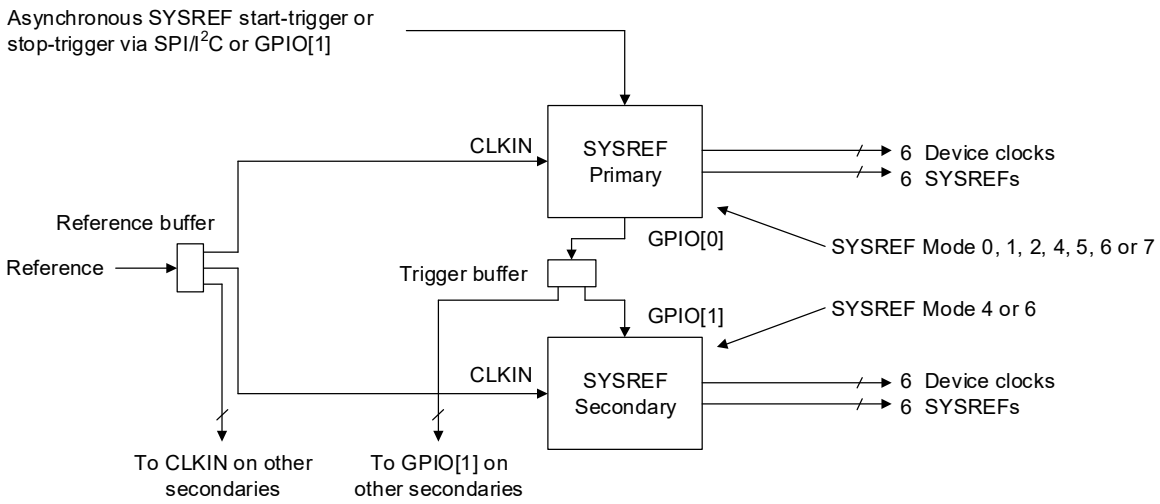


Figure 20. SYSREF Group Using SYSREF Mode 0, 1, 2, 4, 5, 6, or 7

1.11.6.2 Timing for SYSREF Groups Using SYSREF Mode 0, 4, and 5 (Counted)

For SYSREF Mode 0, 4, and 5, the primary device re-times the SYSREF start-trigger that it outputs on GPIO[0], which facilitates the SYSREF events on the primary and secondary devices starting simultaneously. If the sysref_pulse_count register is programmed the same for all devices, then the SYSREF events for all devices in the group can stop simultaneously. See Figure 21 and Figure 22 for SYSREF timing for SYSREF groups using SYSREF Mode 0, 4, and 5.

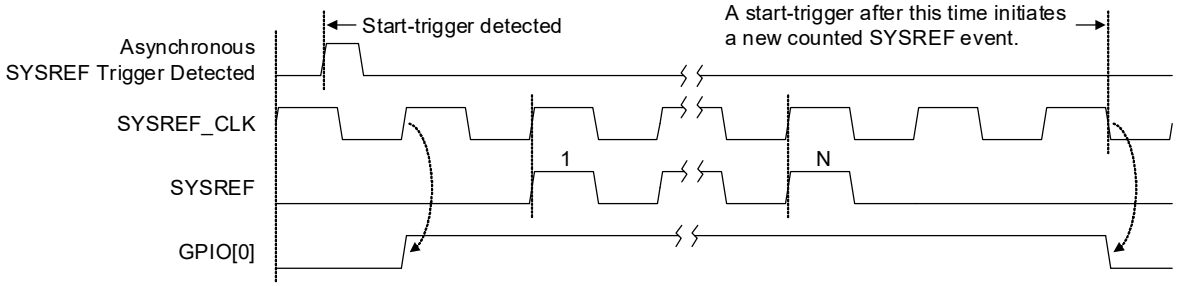


Figure 21. SYSREF Timing for Primary, SYSREF Mode 0, 4, or 5 (Counted)

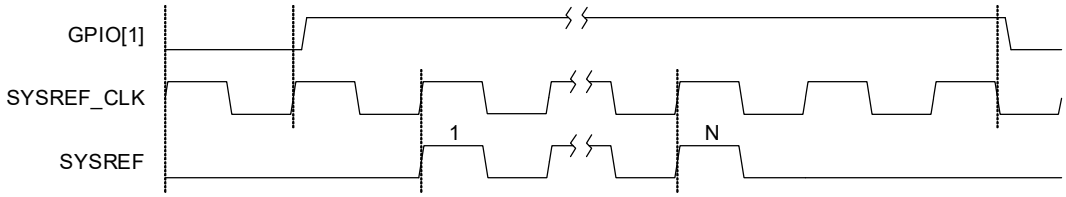


Figure 22. SYSREF Timing for Secondary, SYSREF Mode 4 (Counted)

1.11.6.3 Timing for SYSREF Group Using SYSREF Mode 1, 2, 6, and 7 (Continuous)

For SYSREF Mode 1, 2, 6, and 7, the primary device re-times the SYSREF start-trigger that it outputs on GPIO[0], which facilitates the SYSREF events on the primary and secondary devices starting simultaneously. The primary device does not re-time the SYSREF stop-trigger that it outputs on GPIO[0]. Depending on the timing of the asynchronous stop-trigger received by the primary device, the SYSREF events on the primary and secondary devices may not stop simultaneously. See Figure 23 and Figure 24 for SYSREF timing for SYSREF groups using SYSREF Mode 1, 2, 6, or 7.

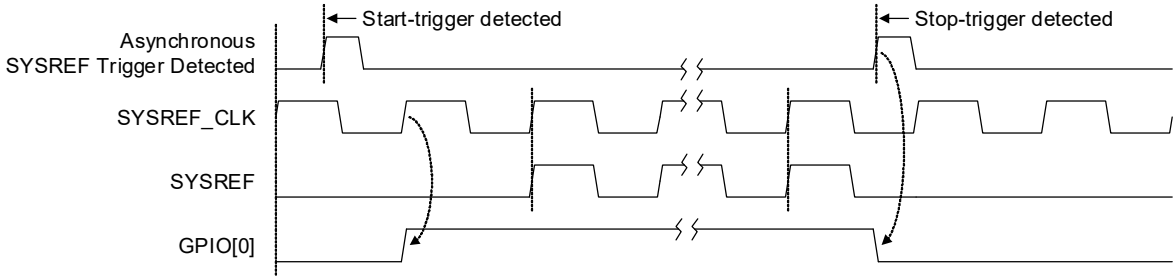


Figure 23. SYSREF Timing for Primary, SYSREF Mode 1, 2, 6 or 7 (Continuous)

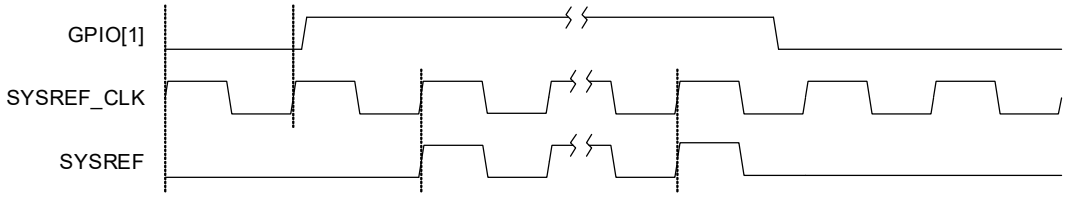


Figure 24. SYSREF Timing for Secondary, SYSREF Mode 6 (Continuous)

1.11.6.4 SYSREF Group Using SYSREF Mode 3

Figure 25 shows a SYSREF group using devices configured for SYSREF mode 3. The primary device accepts an asynchronous SYSREF request and outputs a re-timed SYSREF start-trigger for all devices in the group.

Table 11. Recommended Configurations for Devices in a SYSREF Group Using SYSREF Mode 3

Position in the Group	SYSREF Mode	Recommended Configuration	Comments
Primary or secondary	3	sysref_is_prime = don't care	For SYSREF Mode 3, the primary and secondaries will not output SYSREF start-triggers and stop-triggers on GPIO[0] regardless of the setting of sysref_is_prime.
		sysref_trig_from_prime = 0x0 or 0x1	For SYSREF Mode 3, the primary and secondaries should all use the same setting for sysref_trig_from_prime.
		gpio_func = 0x21 for GPIO[1] pad_gpio_oe_b = 0x1 for GPIO[1]	GPIO[1] will accept SYSREF start-triggers.

The primary device is configured to output the SYSREF_CLK (SYSREF_CLK_OUT) on OUT[x]. The asynchronous SYSREF request is a register write or GPIO input that causes OUT[x] to be synchronously enabled.

The synchronous SYSREF start-trigger for all devices in the group is the first rising edge of SYREF_CLK_OUT that reaches GPIO[1] on each device. See Clock Output Enable for information on how to synchronously enable an OUT[x].

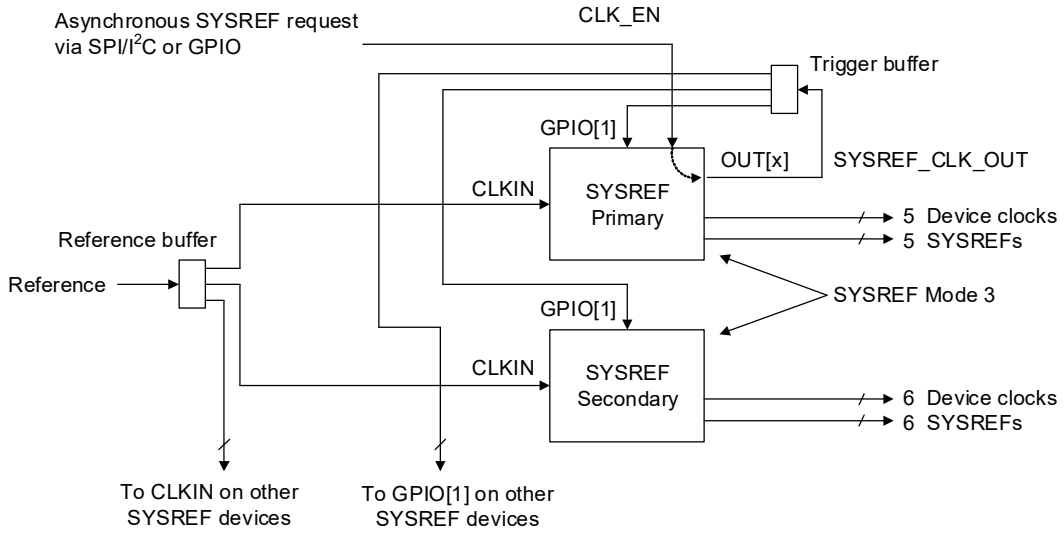


Figure 25. SYSREF Group Using SYSREF Mode 3

Figure 26 shows how the primary and secondary devices react to the SYSREF start-trigger. For a SYSREF group using SYSREF mode 3, all devices in the group should use the same setting for sysref_trig_from_prime to ensure they all react the same to the SYSREF start-trigger.

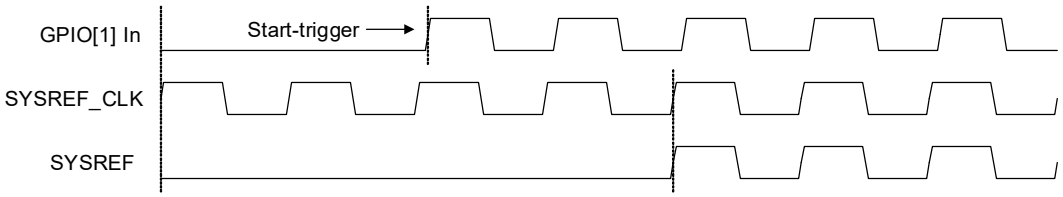


Figure 26. SYSREF Timing for Primary and Secondary, SYSREF Mode 3 with sysref_trig_from_prime = 0x0

The time delay to synchronously enable an OUT[x] can be more than four cycles of the SYSREF_OUT clock. If more precise control of the timing is required, then OUT[x] can be permanently enabled and the trigger buffer in Figure 25 can be replaced with a buffer that has a synchronous clock enable capability, such as the ICS8305.

The SYSREF stop-trigger for SYSREF mode 3 is a low-to-high transition on the sysref_clear_sequential register bit that must be accessed via SPI/I2C for each device.

1.12 Status and Control

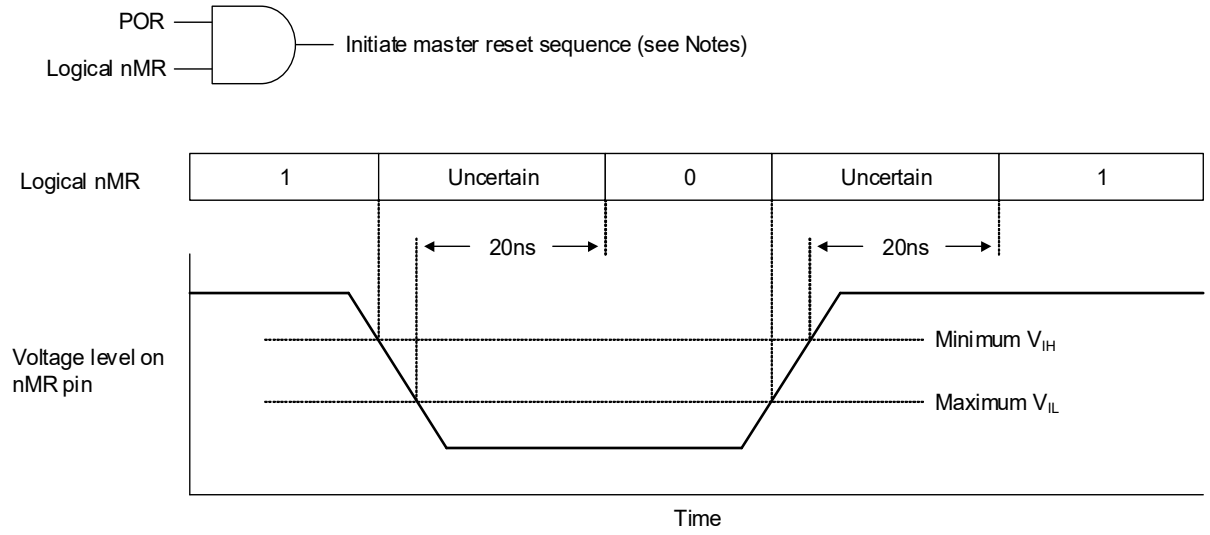
All control and status registers are accessed through a 1MHz I2C or 20MHz SPI slave microprocessor interface. The device can automatically load a configuration from internal one time programmable (OTP) memory. Alternatively, the I2C master interface can automatically load a configuration from an external EEPROM after reset.

2. Power-On Reset and Reset Controller

There are no power supply sequencing requirements; however, if V_{DDOx} or V_{DD_CLK} reach 90% of V_{DD} nominal after V_{DD_DIG} then a soft reset or a master reset must be initiated to ensure the input dividers and output dividers are synchronized. Until V_{DD_DIG} reaches 90% of its nominal voltage, all differential clock outputs are held in tri-state mode. A soft reset can be initiated by setting the self clearing soft_reset register bit to 0x1; a master reset can be initiated as described in the following paragraphs.

Upon power-up, an internal power-on reset (POR) signal is asserted 20ms after the V_{DDXO}, V_{DD_DCD} and V_{DDD33_DIG} supplies all reach 90% of V_{DD} nominal. The first master reset sequence is initiated when POR is asserted and the voltage level on the nMR pin is high.

After the first master reset sequence is initiated, another master reset sequence can be initiated by taking the voltage level on the nMR pin low and then high while POR remains asserted (see Figure 27). To ensure a master reset sequence is initiated, the voltage level on the nMR pin must be held low for at least 20ns before transitioning high. To ensure deterministic behavior, voltage level transitions on the nMR pin must be monotonic between minimum V_{IH} and maximum V_{IL} (see GPIO, Serial Port, and nMR DC Electrical Characteristics in the device datasheet).



Notes
* Requires 1 from logical nMR for the first master reset sequence
* Requires 0 to 1 transition from logical nMR after the first master reset sequence has been initiated

Figure 27. Master Reset Sequence Initiation

The nMR pin has an internal pull-up that can be left to float, or it can optionally be externally pulled high or low. If nMR is high when the internal POR is asserted the reset controller will initiate a master reset sequence. If nMR is low when the internal POR is asserted the reset controller will not initiate a master reset sequence until nMR is taken high.

When a configuration is loaded from EEPROM, the voltage level on the nMR pin must be held high from the time a master reset sequence is initiated until after the EEPROM transactions have completed, as indicated when the device_ready_sts bit is set to 1. Any GPIO can be configured to indicate the state of the device_ready_sts bit.

During the master reset sequence all clock outputs are initially disabled and are enabled during the reset sequence at a point depending on the out_startup bits. Disabled differential outputs will have OUTx = low and nOUTx = high. Disabled LVCMOS outputs with out_cmos_same_phase = 0x1 will have OUTx = low and nOUTx = low. Disabled LVCMOS outputs with out_cmos_same_phase = 0x0 will have OUTx = low and nOUTx = high.

2.1 Master Reset Sequence

The master reset sequence executes the following steps (in the order listed):

1. Latch the levels on the following pins: GPIOs, nCS_A0, SDO_A1, SDA_SDO, and SCL_SCLK.
2. Load the defaults or configuration from OTP memory and EEPROM (if present)
   a. Set the device_ready_sts bit to 0x1
3. Calibrate the VCO
4. Lock the APLL
5. Calibrate the digitally controlled delays (DCD)
6. Synchronize all dividers
7. Start the DPLL state machines

The serial ports are accessible when the device is ready (i.e., device_ready_sts = 0x1).

The device can be configured by the config_sel bits to select an OTP configuration using the voltage levels latched at start-up on the GPIO, nCS_A0, SDO_A1, SDA_SDIO, and SCL_SCLK pins. In addition, the device can be configured by the i2c_addr_sel bits to select the I^2C address using the voltage levels latched at start-up on the GPIO, nCS_A0, and SDO_A1 pins. For pins used in this way, the voltage levels externally applied must not change from the time a master reset or soft reset sequence is initiated until after the device is ready (i.e., device_ready_sts = 0x1).

2.2 Soft Reset Sequence

After the device is ready (i.e., device_ready_sts = 0x1), a soft reset sequence can be initiated by writing 0x1 to the self clearing soft_reset register bit. A soft reset will clear any status bits that are interrupt sources.

The soft reset sequence executes the following steps (in the order listed):

1. If relatch_inputs = 0x1
   b. Latch the levels on the following pins: GPIOs, nCS_A0, SDO_A1, SDA_SDO, and SCL_SCLK.
2. If otp_load_on_soft_reset_en = 0x1
   a. Clear the device_ready_sts bit
   b. Load configuration from OTP memory and EEPROM (if present).
   c. Set the device_ready_sts bit to 0x1
3. Calibrate the VCO
4. Lock the APLL
5. Calibrate the digitally controlled delays (DCD)
6. Synchronize all dividers
7. Start the DPLL state machines.

The reset sequence can be executed starting at Step 3 (Calibrate the VCO) without initiating a soft reset by writing 0x1 to the self clearing apll_reinit register bit.

3. Serial Interfaces

I²C or SPI operation is selected by the ssi_enable register field which defaults to I²C mode. The serial interfaces are inactive until the OTP load completes during the power-up sequence.

3.1 Paging

You can choose to operate the serial port providing the full offset address within each burst, or to operate in a paged mode where part of the address offset is provided in each transaction and another part comes from an internal page register in each serial port. Figure 28 shows how page register and offset bytes from each serial transaction interact to address a register within the RC38312, RC38112, RC38208, RC38108.

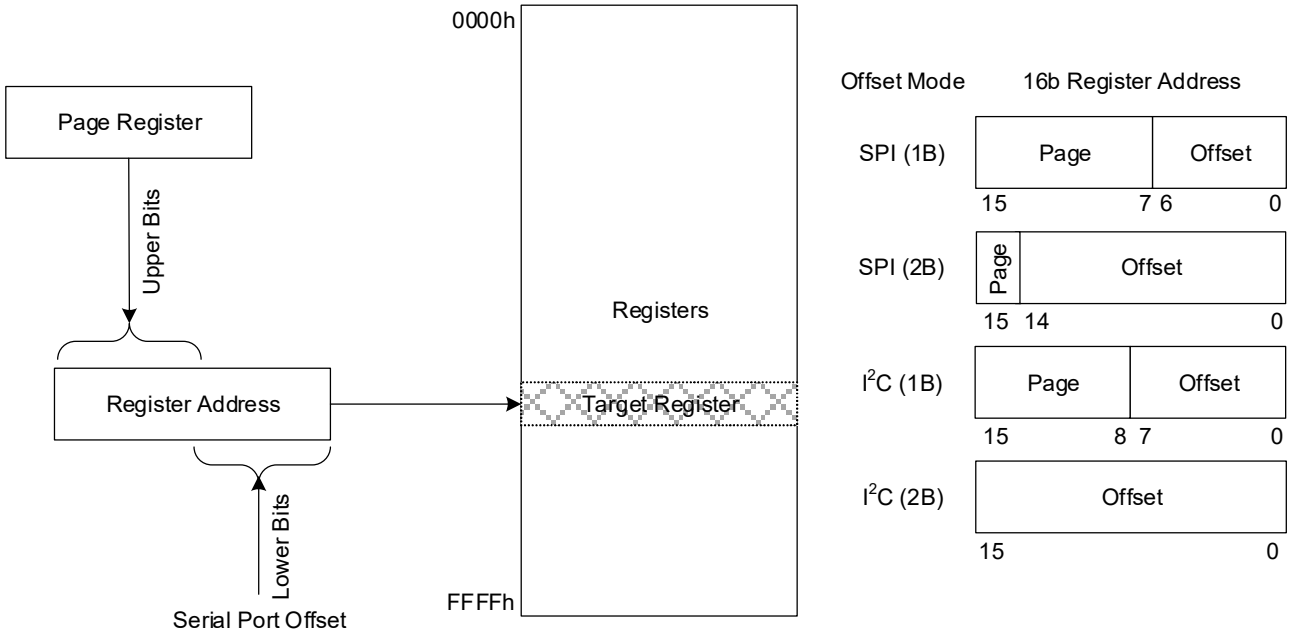


Figure 28. Register Addressing Modes Using Serial Port

3.2 I²C Slave

The I²C slave protocol of the RC38312, RC38112, RC38208, RC38108 complies with the I²C specification, version UM10204 Rev.6 – 4 April 2014. In the following description, serial clock line (SCL) refers to the SCL_SCLK pin and serial data line (SDA) refers to the SDA_SDIO pin.

Figure 29 shows the sequence of states on the I²C SDA signal for the supported modes of operation.

The Dev Addr shown in the figure represents the I²C bus address of the device; this 7-bit value in the i2c_addr register field defaults to 0x09. To use a different I²C bus address, an OTP configuration must alter the value of one or both of the i2c_addr and i2c_addr_sel register fields. The value in the i2c_addr_sel register field determines if the I²C bus address can be controlled through the nCS_A0, SDO_A1 or GPIO pins.

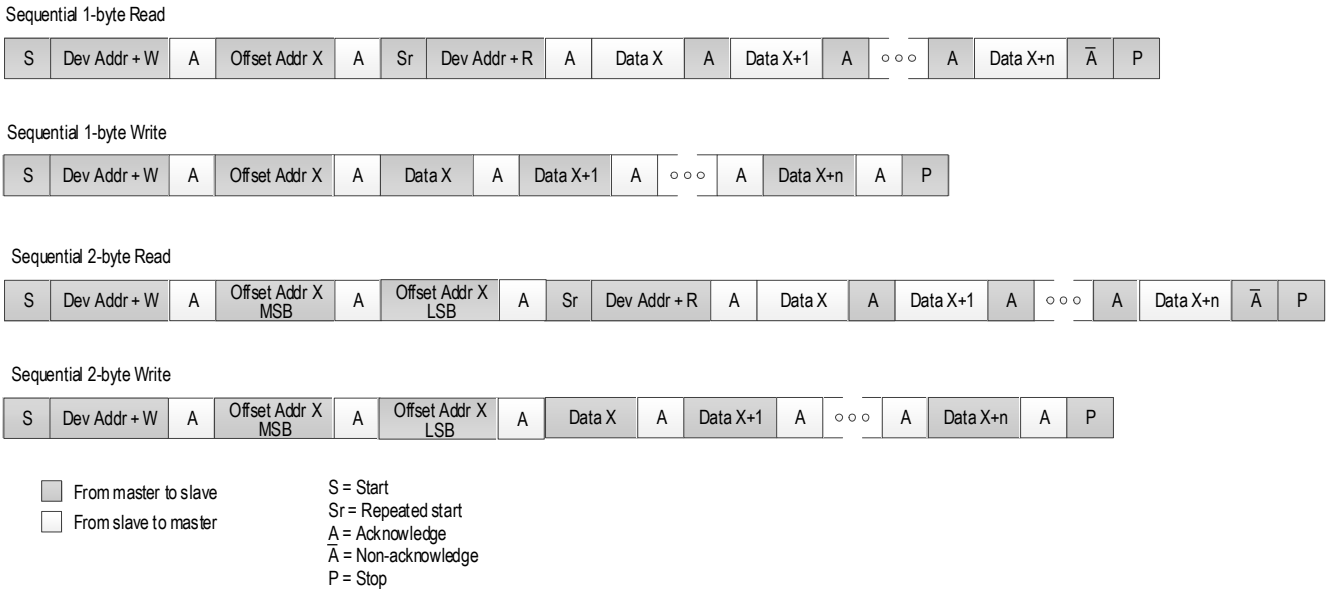


Figure 29. I²C Slave Sequencing

The selection of 1-byte (1B) or 2-byte (2B) offset addressing must also be configured using the ssi_addr_size register field. These offsets are used in conjunction with the page register to access registers internal to the device (see Figure 28). Because the I²C protocol already includes a read/write bit with the Dev Addr, all bits of the 1B or 2B offset field can be used to address internal registers.

- In 1B mode, the lower 8 bits of the register offset address come from the Offset Addr byte and the upper 8 bits come from the page register. The page register can be accessed at any time using an offset byte value of 0xFC. This 4-byte register must be written in a single-burst write transaction.
- In 2B mode, the full 16-bit register address can be obtained from the Offset Addr bytes.

Note: I²C burst mode operation is recommended to ensure data integrity of multi-byte registers. When accessing a multi-byte register, all data bytes must be written or read in a single I²C burst access. Bursts can be of greater length if required but must not extend beyond the end of the register page (Offset Addr 0xFF in 1B mode). An internal address pointer is incremented automatically as each data byte is written or read.

I²C interface timing is shown in the device datasheet.100kHz (Standard mode), 400kHz (Fast mode), and 1MHz (Fast mode plus) operation are supported. The output slew rate is set according to the speed selected by the pad_scl_sclk_drv register field.

The I²C interface operating at 1MHz supports a DCO update rate of approximately 16k updates per second.

3.2.1 I²C 1-byte (1B) Addressing Example

The I²C 7-bit I²C address for RC38312, RC38112, RC38208, RC38108 is 0x09 with LSB = R/W

Example write 0x8003 to register 0x20:

```
12* FC 00 00 00 00      #Set Page Register, *I2C Address is left-shifted one bit.
12 20 03 80             #Write data 0x8003 to 0x20
```

Example read from register 0x168

```
12* FC 00 01 00 00      #Set Page Register, *I2C Address is left-shifted one bit.
12 68                   #Set I2C pointer to 0x168, I2C instruction should use "No Stop".
13 <read back data>     #Send address with Read bit set.
```

3.2.2 I²C 2-byte (2B) Addressing Example

The I²C 7-bit I²C address for RC38312, RC38112, RC38208, RC38108 is 0x09 with LSB = R/W

Example write 0x8003 to register 0x20:

```
12 00 20 03 80          #Write data 0x8003 to 0x0020
```

Example read from register 0x168:

```
12 01 68                #Set I2C pointer to 0x0168, * I2C instruction should use "No Stop".
13 <read back data>     #Send address with Read bit set.
```

3.3 SPI Slave

In the following description, nCS refers to the nCS_A0 pin, SCLK refers to the SCL_SCLK pin, SDI SDIO refer to the SDA_SDIO pin, and SDO refers to the SDO_A1 pin.

The RC38312, RC38112, RC38208, RC38108 supports 4-wire or 3-wire SPI operation as a selectable protocol on the serial port. The 3-wire or 4-wire mode is selected by the spi_3wire register bit. In 4-wire mode, there are separate data in (to the RC38312, RC38112, RC38208, RC38108) and data out signals (SDI and SDO respectively). In 3-wire mode, the SDIO signal is used as a single, bidirectional data signal.

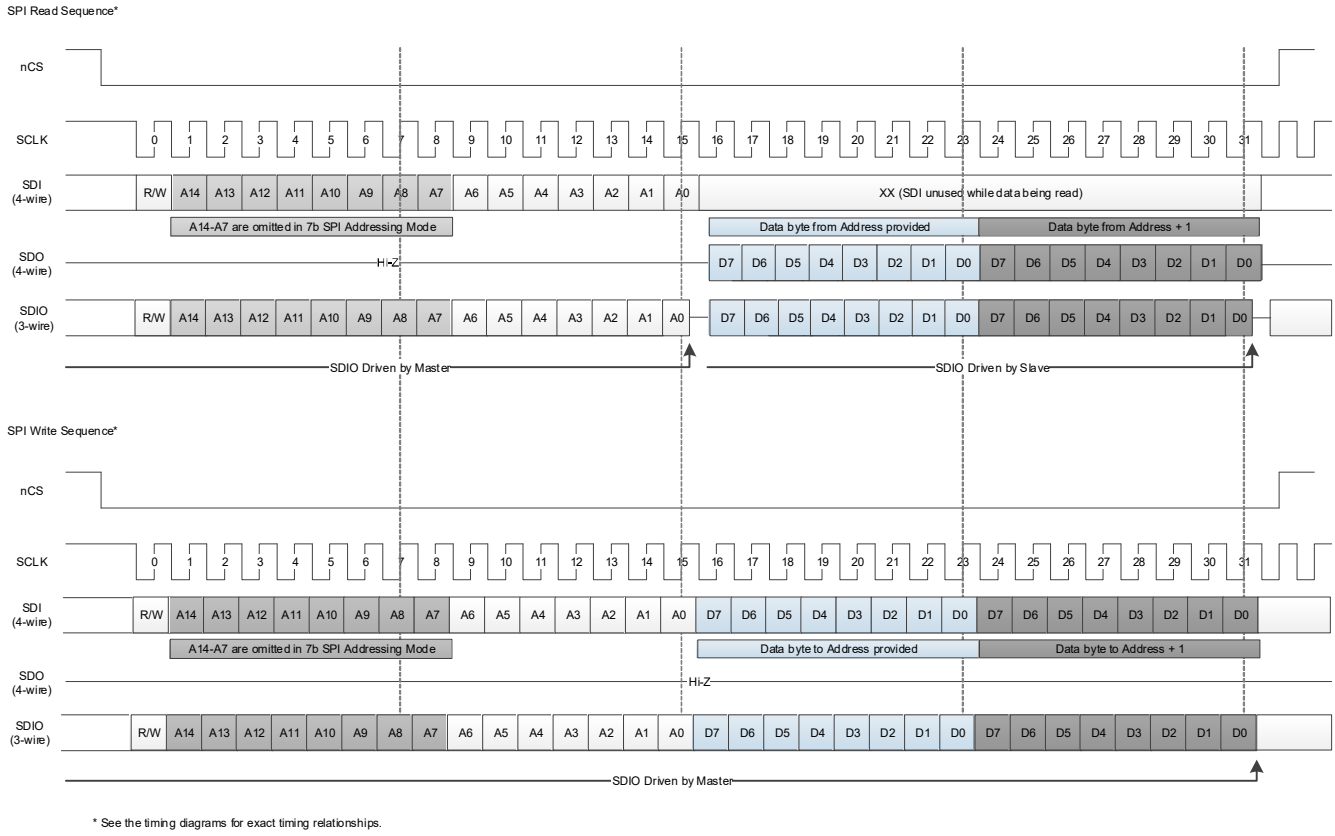


Figure 30. SPI Sequencing

Figure 30 shows the sequencing of address and data on the serial port in both 3-wire and 4-wire SPI mode. 4-wire SPI mode is the default. The R/W bit is high for read cycles and low for write cycles.

SPI operation can be configured for the following settings through register fields:

- 1-byte (1B) or 2-byte (2B) offset addressing (ssi_addr_size) (see Figure 28)
- In 1B operation, the 16-bit register address is formed by using the 7 bits of address supplied in the SPI access and taking the upper 9 bits from the page register. The page register is accessed using an Offset Address of 0x7C with a 4-byte burst access.
- In 2B operation, the 16-bit register address is formed by using the 15 bits of address supplied in the SPI access and the upper 1-bit is fixed to b'0.
- Data sampling on falling or rising edge of SCLK (spi_clk_sel)

- Output (read) data positioning relative to active SCLK edge (spi_del_out)

Note: SPI burst mode operation is recommended to ensure data integrity of multi-byte registers. When accessing a multi-byte register, all data bytes must be written or read in a single SPI burst access. Bursts can be of greater length if desired but must not extend beyond the end of the register page. An internal address pointer is incremented automatically as each data byte is written or read.

SPI timing is shown in the device datasheet.

The SPI interface operating at 20MHz supports a DCO update rate of approximately 400k updates per second.

3.3.1 SPI 1-byte (1B) Addressing Example

Example write to “50” to register 0xE4:

```
7C 80 00 00 00          #Set Page register
64* 50                  #*MSB is 0 for write transactions
```

Example read from 0x24:

```
7C 00 00 00 00          #Set Page register
A4* 00                  #*MSB is set, so this is a read command
```

3.3.2 SPI 2-byte (2B) Addressing Example

Example write to “50” to register 0xCBE4

```
4B E4* 50               #*MSB is 0 for write transactions
```

Example read from 0xC024:

```
C0* 24 00               #*MSB is set, so this is a read command
```

4. Register Set Descriptions

Table 12. Register Set Module Index

Module Base Address (Hex)	Name	Module Description	Link
0x0	GLOBAL	Global Control and Status Registers	GLOBAL
0x40	INT	Interrupt Registers	INT
0x50	SSI	Slave Serial Interface Registers	SSI
0x60	XO	Crystal Oscillator, Input Buffer and Reference Select Registers	XO
0x80	APLL	APLL Registers	APLL
0x100	OUTBUF[0]	Output Buffer Registers	OUTBUF
0x108	OUTBUF[1]	Same as OUTBUF[0]	OUTBUF
0x110	OUTBUF[2]	Same as OUTBUF[0]	OUTBUF
0x118	OUTBUF[3]	Same as OUTBUF[0]	OUTBUF
0x120	OUTBUF[4]	Same as OUTBUF[0]	OUTBUF
0x128	OUTBUF[5]	Same as OUTBUF[0]	OUTBUF
0x130	OUTBUF[6]	Same as OUTBUF[0]	OUTBUF
0x138	OUTBUF[7]	Same as OUTBUF[0]	OUTBUF
0x140	OUTBUF[8]	Same as OUTBUF[0]	OUTBUF
0x148	OUTBUF[9]	Same as OUTBUF[0]	OUTBUF
0x150	OUTBUF[10]	Same as OUTBUF[0]	OUTBUF
0x158	OUTBUF[11]	Same as OUTBUF[0]	OUTBUF
0x160	SYSDIV	System Clock Divider Registers	SYSDIV
0x170	IOD[0]	Integer Output Divider Registers	IOD
0x180	IOD[1]	Same as IOD[0]	IOD
0x190	IOD[2]	Same as IOD[0]	IOD
0x1A0	IOD[3]	Same as IOD[0]	IOD
0x1B0	IOD[4]	Same as IOD[0]	IOD
0x1C0	IOD[5]	Same as IOD[0]	IOD
0x1D0	IOD[6]	Same as IOD[0]	IOD
0x1E0	IOD[7]	Same as IOD[0]	IOD
0x1F0	IOD[8]	Same as IOD[0]	IOD
0x200	IOD[9]	Same as IOD[0]	IOD
0x210	IOD[10]	Same as IOD[0]	IOD
0x220	IOD[11]	Same as IOD[0]	IOD
0x230	SYSREF	SYSREF Registers	SYSREF
0x240	GPIO[0]	General Purpose IO Registers Instances 0-7 apply to GPIO0 to GPIO9 Instance 8 applies to LOCK	GPIO
0x248	GPIO[1]	Same as GPIO[0]	GPIO
0x250	GPIO[2]	Same as GPIO[0]	GPIO
0x258	GPIO[3]	Same as GPIO[0]	GPIO
0x260	GPIO[4]	Same as GPIO[0]	GPIO
0x300	FOD[0]	Fractional Output Divider Registers	FOD
0x340	FOD[1]	Same as FOD[0]	FOD
0x380	FOD[2]	Same as FOD[0]	FOD
0x400	INPUTBUF[0]	Input Buffer Registers	INPUTBUF

Table 12. Register Set Module Index

Module Base Address (Hex)	Name	Module Description	Link
0x402	INPUTBUF[1]	Same as INPUTBUF[0]	INPUTBUF
0x404	INPUTBUF[2]	Same as INPUTBUF[0]	INPUTBUF
0x406	INPUTBUF[3]	Same as INPUTBUF[0]	INPUTBUF
0x410	INPUTMUX	Input Mux Registers	INPUTMUX
0x420	INPUTDIV[0]	Input Buffer Registers	INPUTDIV
0x424	INPUTDIV[1]	Same as INPUTDIV[0]	INPUTDIV
0x428	INPUTDIV[2]	Same as INPUTDIV[0]	INPUTDIV
0x42C	INPUTDIV[3]	Same as INPUTDIV[0]	INPUTDIV
0x430	INPUTOFFSET	Reference Offset Registers	INPUTOFFSET
0x440	TDCAPLL	TDC APLL Registers	TDCAPLL
0x500	DPLL[0]	DPLL Registers	DPLL
0x600	DPLL[1]	Same as DPLL[0]	DPLL
0x700	DPLL[2]	Same as DPLL[0]	DPLL
0x800	XTALMON	XTAL Monitor Registers (xtalmon[0] = XIN, xtalmon[1]=REF4)	XTALMON
0x810	LOSMON[0]	LOS Monitor Registers	LOSMON
0x820	LOSMON[1]	Same as LOSMON[0]	LOSMON
0x830	LOSMON[2]	Same as LOSMON[0]	LOSMON
0x840	LOSMON[3]	Same as LOSMON[0]	LOSMON
0x850	LOSMON[4]	Same as LOSMON[0]	LOSMON
0x860	FREQMON[0]	Frequency Monitor Registers	FREQMON
0x880	FREQMON[1]	Same as FREQMON[0]	FREQMON
0x8A0	FREQMON[2]	Same as FREQMON[0]	FREQMON
0x8C0	FREQMON[3]	Same as FREQMON[0]	FREQMON
0x8E0	EEPROM	EEPROM Registers	EEPROM
0x900	OTP	OTP Registers	OTP
0x950	LDO	LDO Registers	LDO
0xA00	TIME_SYNC_TOD	Timesync TOD and Clock Registers	TIME_SYNC_TOD
0xA80	TIME_SYNC_LPF	Timesync LPF Registers	TIME_SYNC_LPF
0xB00	TIME_SYNC_TDC	Timesync TDC Registers	TIME_SYNC_TDC
0xC00	TIME_SYNC_TDC_FIFO_DBG	Timesync TDC FIFO Debug Registers	TIME_SYNC_TDC_FIFO_DBG
0xD00	REINIT	THIS SECTION CONSISTS OF ALIASES TO THE GLOBAL CONTROL REGISTERS	REINIT

4.1 GLOBAL

Global Control and Status Registers.

Table 13. GLOBAL Register Index

Offset (Hex)	Register Module Base Address: 0x0	
	Register Name	Register Description
0x0	VENDOR_ID	Vendor ID and Device Type
0x2	DEVICE_ID	Device ID
0x4	DEVICE_REV	Device Revisions and Font ID
0x6	DEVICE_PGM	Device Dash Code

Table 13. GLOBAL Register Index

Offset (Hex)	Register Module Base Address: 0x0	
	Register Name	Register Description
0x8	DEVICE_CNFG	Device Configuration
0xC	MISC_CNFG	Reset Configuration
0x10	SCRATCH_CNFG	Scratch register
0x14	OE_CTRL	Output Enable Control
0x15	SOFT_RESET_CTRL	Soft Reset Control
0x16	DIVIDER_SYNC_CTRL	Divider sync control
0x20	STARTUP_STS	Startup status
0x22	TEMP_SENSOR_STS	Temperature Sensor Value
0x24	DEVICE_STS	Device status

4.1.1 VENDOR_ID

Vendor ID and Device Type.

VENDOR_ID Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:12	dev_id_type	RO	0x1	Device ID Block Type. A value of 0x1 indicates that this register is followed by a 16-bit Device ID register and an 16-bit Device Revision register, and a 16-bit Device Programming register.
11	reserved	RO	0x0	Reserved
10:0	vendor_id	RO	0x33	Vendor ID. IDT JEDEC ID.

4.1.2 DEVICE_ID

Device ID.

DEVICE_ID Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:0	device_id	RW	0x0	Device ID. For default value refer to Product Identification. This field is write-able so it may be configured from OTP.

4.1.3 DEVICE_REV

Device Revisions and Font ID.

DEVICE_REV Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:13	reserved	RO	0x0	Reserved
12:8	font_id	RO	0x6	Font ID. Font ID to distinguish die variants. Decode as follows: 0x1 = Font 001 0x2 = Font 002 0x3 = Font 003 0x4 = Font 004 0x5 = Font 005 0x6 = Font 006 0x7 = Font 007
7:0	reserved	RO	0x62	Reserved

4.1.4 DEVICE_PGM

Device Dash Code.

DEVICE_PGM Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:0	dash_code	RW	0x0	Dash code. Decimal value assigned by Renesas to identify the user configuration loaded in OTP at the factory. This field is write-able and is configured from the OTP common configuration programmed at the factory. 0x0 = No user configurations are programmed at the factory

4.1.5 DEVICE_CNFG

Device Configuration.

DEVICE_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:12	device_configuration	RW	0x0	Device configuration. This field is unused in test vehicle.
11:8	eeprom_addr_sel	RW	0x0	EEPROM Address Select for bits[2:0]. Controls source of the external EEPROM device address for bits [2:0], from a combination of csrs and pins at startup. gpio[0], gpio[1], and gpio[2] refers to any GPIO pin which has gpio_startup_mode set to 0x5, 0x6, and 0x7 respectively. 0x0 = eeprom_addr[2], eeprom_addr[1], eeprom_addr[0] 0x1 = eeprom_addr[2], eeprom_addr[1], SDO 0x2 = eeprom_addr[2], eeprom_addr[1], nCS 0x3 = eeprom_addr[2], SDO, nCS 0x4 = eeprom_addr[2], gpio[1], gpio[0] 0x5 = gpio[2], eeprom_addr[1], gpio[0] 0x6 = gpio[2], eeproom_addr[1], eeprom_addr[0] 0x7 = gpio[2], SDO, nCS 0x8 = gpio[2], gpio[1], SDO 0x9 = gpio[2], gpio[1], nCS
eeprom_addr_sel (continued)				0xA = Reserved 0xB = eeprom_addr[2], SDO, gpio[0] 0xC = Reserved 0xD = SDO, gpio[1], gpio[0] 0xE = nCS, gpio[1], gpio[0] 0xF = gpio[2], gpio[1], gpio[0]
7:4	i2c_addr_sel	RW	0x0	I2C Address Select for bits [2:0]. Controls source of i2c_addr[2:0] from a combination of csrs and pins at startup. gpio[0], gpio[1], and gpio[2] refers to any GPIO pin which has gpio_startup_mode set to 0x2, 0x3, and 0x4 respectively. 0x0 = i2c_addr[2], i2c_addr[1], i2c_addr[0] 0x1 = i2c_addr[2], i2c_addr[1], SDO 0x2 = i2c_addr[2], i2c_addr[1], nCS 0x3 = i2c_addr[2], SDO, nCS 0x4 = i2c_addr[2], gpio[1], gpio[0] 0x5 = gpio[2], i2c_addr[1], gpio[0] 0x6 = gpio[2], i2c_addr[1], i2c_addr[0] 0x7 = gpio[2], SDO, nCS 0x8 = gpio[2], gpio[1], SDO 0x9 = gpio[2], gpio[1], nCS
i2c_addr_sel (continued)				0xA = Reserved 0xB = i2c_addr[2], SDO, gpio[0] 0xC = Reserved 0xD = SDO, gpio[1], gpio[0] 0xE = nCS, gpio[1], gpio[0] 0xF = gpio[2], gpio[1], gpio[0]

DEVICE_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
3:0	config_sel	RW	0x0	OTP Config Select. Controls source of otp config selection from a combination of pins at startup. gpio[0], and gpio[1] refers to any GPIO pin which has gpio_startup_mode set to 0x0, and 0x1 respectively. 0x0 = User config 0 0x1 = User config 1 0x2 = User config 2 0x3 = User config 3 0x4 = SDA_SDIO, SCL_SCLK 0x5 = SDA_SDIO, SDO_A1 0x6 = SDA_SDIO, nCS_A0 0x7 = SDA_SDIO, gpio[0] 0x8 = SCL_SCLK, SDO_A1 0x9 = SCL_SCLK, nCS_A0
config_sel (continued)				0xA = SCL_SCLK, gpio[0] 0xB = SDO_A1, gpio[0] 0xC = nCS_A0, SDO_A1 0xD = nCS_A0, gpio[0] 0xE = gpio[1], gpio[0] 0xF = SDO_A1, nCS_A0

4.1.6 MISC_CNFG

Reset Configuration.

MISC_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:12	reserved	RO	0x0	Reserved
11	temp_sensor_en	RW	0x0	Analog temperature sensor enable. When set, enables the analog temperature sensor used for the DPLL temperature phase offset compensation.
10:8	otp_load_delay	RW	0x1	OTP load delay during startup. Selects wait time for during otp load. Applies even for empty otp images. This is not a timeout. OTP timeout occurs at 25 ms 0x0 = 1 ms 0x1 = 250 us 0x2 = 500 us 0x3 = 2.5 ms 0x4 = 5 ms 0x5 = 10 ms 0x6 = 20 ms 0x7 = Reserved
7	otp_load_on_soft_reset_en	RW	0x0	When set, will load the full OTP on a soft reset
6	relatch_inputs	RW	0x0	Relatch inputs on soft reset (RevA feature). Selects whether or not to relatch pin states when performing a soft reset 0x0 = Don’t relatch inputs on soft reset 0x1 = Relatch inputs on soft reset
5	disable_out_on_resync	RW	0x1	Disable outputs upon resync. When set, whenever a divider sync is requested (either to single dividers or chip-wide), the corresponding outputs will have their OE disabled during the sync process.
4	byp_apll_lock_startup	RW	0x0	Bypass APLL lock at startup. When set, the control state machine does not wait for APLL lock during the boot sequence.

MISC_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
3	byp_wait_ph_adj_done	RW	0x1	Bypass waiting for phase adj done. When set, the control state machine does not wait for the phase adjusts to be done after issuing a divider sync.
2:0	out_startup	RW	0x1	Output enable on startup. Controls the point at which the clock output drivers are enabled during the startup sequence. 0x0 = Clock outputs are disabled until APLL lock asserts (or times out) 0x1 = Clock outputs are disabled until APLL lock asserts (or times out) and DCD and FOD calibration is done 0x2 = Clock outputs are disabled until APLL lock asserts and DCD and FOD calibration is done. Note: This setting can cause the reset controller to remain waiting for APLL lock and never reach its done state. 0x3 = Clock outputs are disabled until APLL lock asserts, DCD calibration is done, and DPLL lock asserts. Note: This setting can cause the reset controller to remain waiting for APLL or DPLL lock and never reach its done state. 0x4 = Clock output drivers are never gated during startup. This setting should only be used when all outputs are disabled with out_driver_en set to 0 0x5 = Reserved 0x6 = Reserved 0x7 = Reserved

4.1.7 SCRATCH_CNFG

Scratch register.

SCRATCH_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:0	scratch	RW	0x0	Scratch register. For arbitrary software use.

4.1.8 OE_CTRL

Output Enable Control.

OE_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	global_oe	RW	0x1	Global Output Enable. This bit is used to control the output enable of all the outputs. When cleared, it overrides all the individual output enable bits and disables all the outputs. When set, the individual output enable settings control the enabling of the output drivers. 0x0 = All outputs are disabled 0x1 = Output enables are controlled by their individual settings

4.1.9 SOFT_RESET_CTRL

Soft Reset Control.

SOFT_RESET_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved

SOFT_RESET_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
1	apll_reinit	RW	0x0	APLL Reinitialization. Writing this bit to 1 re-starts the startup sequence from the VCO calibration step, including divider synchronization.
0	soft_reset	RW	0x0	Soft Reset. Write '1' to trigger a soft reset which re-starts the reset sequence from the VCO calibration step. This bit will self-clear. Depending on the value of latch_inputs, inputs will be relatched to enable changing of the I2C address only (RevA feature).

4.1.10 DIVIDER_SYNC_CTRL

Divider sync control.

DIVIDER_SYNC_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7	vco_iod_sync	RW1S	0x0	Enables the manual resynch of VCO based IODs and DPLL0 FBDIV when it's source is the VCO. Enables the manual resync of VCO based IODs and DPLL0 FBDIV when it's source is the VCO This bit is high until the resync is done.
6:4	fod_iod_sync	RW1S	0x0	Enables the manual resync of FODs and FOD sourced IODs as well as DPLL FBDIVS when source is the FOD. Enables the manual resync of FODs and FOD sourced IODs as well as DPLL FBDIVS when source is the FOD Each bit will remain high until the corresponding FOD has been resynced.
3:1	reserved	RO	0x0	Reserved
0	divider_sync	RW	0x0	Divider Synchronization. Write '1' to trigger synchronization of DPLL and output dividers. The VCO-based IODs and DPLL0 FBDIV (if driven from VCO) are done first, followed by FOD0 and DPLL0 FBDIV (if driven from FOD0), followed by FOD1 and finally by FOD2. This bit auto-clears.

4.1.11 STARTUP_STS

Startup status.

STARTUP_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15	scl_sclk_at_startup_sts	RO	0x0	SCL_SCLK Value Latched at Startup. Value latched at startup, (when a rising edge on nMR is detected) 0x0 = Low 0x1 = High
14	sda_sdio_at_startup_sts	RO	0x0	SDA_SDIO Value Latched at Startup. Value latched at startup, (when a rising edge on nMR is detected) 0x0 = Low 0x1 = High
13	sdo_a1_at_startup_sts	RO	0x0	SDO_A1 Value Latched at Startup. Value latched at startup, (when a rising edge on nMR is detected) 0x0 = Low 0x1 = High
12	ncs_a0_at_startup_sts	RO	0x0	NCS_A0 Value Latched at Startup. Value latched at startup, (when a rising edge on nMR is detected) 0x0 = Low 0x1 = High

STARTUP_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
11:5	reserved	RO	0x0	Reserved
4:0	gpio_at_startup_sts	RO	0x0	GPIOx Value Latched at Startup. Value latched at startup, (when a rising edge on nMR is detected) 0x0 = Low 0x1 = High

4.1.12 TEMP_SENSOR_STS

Temperature Sensor Value.

TEMP_SENSOR_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15	reserved	RO	0x0	Reserved
14:0	temp_sensor_value_sts	RO	0x0	Temperature Sensor Value. Value from the analog temperature sensor. Example values: 15'h7FFF: 110 degC 15'h3FFF: 100 degC 15'h003F: 20 degC 15'h001F: 10 degC 15'h0001: -30 degC 15'h0000: -40 degC

4.1.13 DEVICE_STS

Device status.

DEVICE_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:8	reserved	RO	0x0	Reserved
7	rst_done_sts	RO	0x0	Reset Done. Set to 1 when the reset sequence is either done or in the dpll_lock state.
6	device_ready_sts	RO	0x0	Device Ready. Set to 1 when the configuration load (OTP and/or EEPROM) completes during the startup sequence.
5	eeprom_config_valid_sts	RO	0x0	Valid EEPROM User Configuration Loaded. Indicates that the user configuration in config_loaded was successfully loaded from EEPROM. Only valid when device_ready is 1.
4	otp_config_valid_sts	RO	0x0	Valid OTP User Configuration Loaded. Indicates that the user configuration in config_loaded was successfully loaded from OTP. Only valid when device_ready is 1.
3	otp_crc_err_sts	RO	0x0	OTP CRC Error. Indicates that an OTP CRC error was detected. Only valid when device_ready is 1.
2:0	config_loaded_sts	RO	0x0	User Configuration Loaded. Indicates the user configuration loaded from OTP/EEPROM on start-up or an OTP reload. Note that on startup, the common configuration is always loaded prior to the user configuration. Only valid when device_ready is 1. 0 = Common config 1 = User config 0 2 = User config 1 3 = User config 2 4 = User config 3

4.2 INT

Interrupt Registers.

Table 14. INT Register Index

Offset (Hex)	Register Module Base Address: 0x40	
	Register Name	Register Description
0x0	INT_EN_CTRL	Interrupt Enable Configuration
0x8	INT_STS	Interrupt Status

4.2.1 INT_EN_CTRL

Interrupt Enable Configuration.

INT_EN_CTRL Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
63:47	reserved	RO	0x0	Reserved
46	ts_tdc_fifo_overrun_int_en	RW	0x0	TDC FIFO Overrun interrupt enable. When this field is set to 1, the ts_tdc_fifo_overrun_int_sts bit contributes to the device interrupt.
45	ts_tdc_fifo_alarm_int_en	RW	0x0	TDC FIFO Alarm interrupt enable. When this field is set to 1, the ts_tdc_fifo_alarm_int_sts bit contributes to the device interrupt.
44	ts_tod_compare_int_en	RW	0x0	TOD Compare interrupt enable. When this field is set to 1, the ts_tod_compare_int_sts bit contributes to the device interrupt.
43	i2c_crc_err_int_en	RW	0x0	I2C CRC ERROR interrupt enable. When this field is set to 1, the i2c_crc_int_sts bit contributes to the device interrupt.
42:40	dpll_bw_sel_int_en	RW	0x0	Manual bandwidth select interrupt enable. When this field is set to 1, the bw_sel_int_sts bit contributes to the device interrupt.
39	load_fail_int_en	RW	0x0	Configuration Loader Failure Interrupt Enable. When this field is set to 1, the load_fail_int_sts bit contributes to the device interrupt.
38	load_err_int_en	RW	0x0	Configuration Loader Error Interrupt Enable. When this field is set to 1, the load_err_int_sts bit contributes to the device interrupt.
37	reserved	RW	0x0	Reserved
36	xtal_lmt_int_en	RW	0x0	XTAL Monitor LOS Threshold Exceeded interrupt enable. When this field is set to 1, the xtal_lmt_int_sts bit contributes to the device interrupt.
35	los4_lmt_int_en	RW	0x0	Ref4 Monitor LOS Threshold Exceeded interrupt enable. When this field is set to 1, the los4_lmt_int_sts bit contributes to the device interrupt.
34	los3_lmt_int_en	RW	0x0	Ref3 Monitor LOS Threshold Exceeded interrupt enable. When this field is set to 1, the los3_lmt_int_sts bit contributes to the device interrupt.
33	los2_lmt_int_en	RW	0x0	Ref2 Monitor LOS Threshold Exceeded interrupt enable. When this field is set to 1, the los2_lmt_int_sts bit contributes to the device interrupt.
32	los1_lmt_int_en	RW	0x0	Ref1 Monitor LOS Threshold Exceeded interrupt enable. When this field is set to 1, the los1_lmt_int_sts bit contributes to the device interrupt.
31	los0_lmt_int_en	RW	0x0	Ref0 Monitor LOS Threshold Exceeded interrupt enable. When this field is set to 1, the los0_lmt_int_sts bit contributes to the device interrupt.
30:28	dpll_lol_lmt_int_en	RW	0x0	DPLL Loss-of-lock Threshold Exceeded interrupt enable. When this field is set to 1, the dpll_lol_lmt_int_sts bit contributes to the device interrupt.
27	apll_lol_lmt_int_en	RW	0x0	APLL Loss-of-lock Threshold Exceeded interrupt enable. When this field is set to 1, the apll_lol_lmt_int_sts bit contributes to the device interrupt.
26	freq3_update_int_en	RW	0x0	Ref3 Frequency Monitor Offset Valid interrupt enable. When this field is set to 1, the freq3_update_int_sts bit contributes to the device interrupt.
25	freq2_update_int_en	RW	0x0	Ref2 Frequency Monitor Offset Valid interrupt enable. When this field is set to 1, the freq2_update_int_sts bit contributes to the device interrupt.
24	freq1_update_int_en	RW	0x0	Ref1 Frequency Monitor Offset Valid interrupt enable. When this field is set to 1, the freq1_update_int_sts bit contributes to the device interrupt.

INT_EN_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
23	freq0_update_int_en	RW	0x0	Ref0 Frequency Monitor Offset Valid interrupt enable. When this field is set to 1, the freq0_update_int_sts bit contributes to the device interrupt.
22	freq3_int_en	RW	0x0	Ref3 Frequency Monitor interrupt enable. When this field is set to 1, the freq3_int_sts bit contributes to the device interrupt
21	freq2_int_en	RW	0x0	Ref2 Frequency Monitor interrupt enable. When this field is set to 1, the freq2_int_sts bit contributes to the device interrupt.
20	freq1_int_en	RW	0x0	Ref1 Frequency Monitor interrupt enable. When this field is set to 1, the freq1_int_sts bit contributes to the device interrupt.
19	freq0_int_en	RW	0x0	Ref0 Frequency Monitor interrupt enable. When this field is set to 1, the freq0_int_sts bit contributes to the device interrupt.
18	xtal_int_en	RW	0x0	XTAL Monitor Loss-of-Signal interrupt enable. When this field is set to 1, the xtal_int_sts bit contributes to the device interrupt.
17	los4_int_en	RW	0x0	Ref4 Monitor Loss-of-Signal interrupt enable. When this field is set to 1, the los4_int_sts bit contributes to the device interrupt.
16	los3_int_en	RW	0x0	Ref3 Monitor Loss-of-Signal interrupt enable. When this field is set to 1, the los3_int_sts bit contributes to the device interrupt.
15	los2_int_en	RW	0x0	Ref2 Monitor Loss-of-Signal interrupt enable. When this field is set to 1, the los2_int_sts bit contributes to the device interrupt.
14	los1_int_en	RW	0x0	Ref1 Monitor Loss-of-Signal interrupt enable. When this field is set to 1, the los1_int_sts bit contributes to the device interrupt.
13	los0_int_en	RW	0x0	Ref0 Monitor Loss-of-Signal interrupt enable. When this field is set to 1, the los0_int_sts bit contributes to the device interrupt.
12:10	dpll_state_ch_int_en	RW	0x0	DPLL State Change interrupt enable. When this field is set to 1, the dpll_state_ch_int_sts bit contributes to the device interrupt.
9:7	dpll_holdover_int_en	RW	0x0	DPLL Holdover interrupt enable. When this field is set to 1, the dpll_holdover_int_sts bit contributes to the device interrupt.
6:4	dpll_lol_int_en	RW	0x0	DPLL Loss-of-Lock interrupt enable. When this field is set to 1, the dpll_lol_int_sts bit contributes to the device interrupt.
3:2	reserved	RW	0x0	Reserved
1	apll_lock_int_en	RW	0x0	APLL Lock Interrupt enable. When this field is set to 1, the apll_lock_int_sts bit contributes to the device interrupt.
0	apll_lol_int_en	RW	0x0	APLL Loss-of-Lock interrupt enable. When this field is set to 1, the apll_lol_int_sts bit contributes to the device interrupt.

4.2.2 INT_STS

Interrupt Status.

INT_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:47	reserved	RO	0x0	Reserved
46	ts_tdc_fifo_overrun_int_sts	RO	0x0	TDC FIFO Overrun Interrupt Status. When set, indicates that the Time Sync TDC FIFO overrun has occurred. This bit reflects the fifo_overrun_evt.
45	ts_tdc_fifo_alarm_int_sts	RO	0x0	TDC FIFO Alarm Interrupt Status. When set, indicates that the Time Sync TDC FIFO count alarm has triggered. This bit reflects the fifo_count_alarm_evt.
44	ts_tod_compare_int_sts	RO	0x0	TOD Compare Interrupt Status. When set, indicates that the Time Sync target TOD value has been reached. This bit reflects the tod_compare_evt.
43	i2c_crc_err_int_sts	RO	0x0	I2C CRC ERROR interrupt status. When set, indicates that an I2C CRC error has occurred. This bit reflects the i2c_crc_err_evt.
42:40	dpll_bw_sel_int_sts	RO	0x0	Manual bandwidth selection Interrupt Status. When set, indicates that the DPLL bw_sel has changed. This bit reflects the dpll_bw_sel_ch_evt.

INT_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
39	load_fail_int_sts	RO	0x0	Configuration Loader Failure interrupt status. When set, indicates that the OTP or EEPROM load failed. This bit is the logical OR of the otp_load_fail_evt and eeprom_load_fail_evt bits.
38	load_err_int_sts	RO	0x0	Configuration Loader Error interrupt status. When set, indicates that the OTP or EEPROM detected a CRC error. This bit is the logical OR of the otp_crc_err_evt and eeprom_crc_err_evt event bits.
37	reserved	RO	0x0	Reserved
36	xtal_lmt_int_sts	RO	0x0	XTAL Monitor LOS Threshold Exceeded interrupt status. When set, indicates that the number of times XTALMON declared LOS exceeded the programmed limit. This bit reflects the xtal_los_lmt_evt.
35	los4_lmt_int_sts	RO	0x0	Ref4 Monitor LOS Threshold Exceeded interrupt status. When set, indicates that the number of times LOSMON4 declared LOS exceeded the programmed limit. This bit reflects the los4_los_lmt_evt.
34	los3_lmt_int_sts	RO	0x0	Ref3 Monitor LOS Threshold Exceeded interrupt status. When set, indicates that the number of times LOSMON3 declared LOS exceeded the programmed limit. This bit reflects the losmon3_los_lmt_evt.
33	los2_lmt_int_sts	RO	0x0	Ref2 Monitor LOS Threshold Exceeded interrupt status. When set, indicates that the number of times LOSMON2 declared LOS exceeded the programmed limit. This bit reflects the losmon2_los_lmt_evt.
32	los1_lmt_int_sts	RO	0x0	Ref1 Monitor LOS Threshold Exceeded interrupt status. When set, indicates that the number of times LOSMON1 declared LOS exceeded the programmed limit. This bit reflects the losmon1_los_lmt_evt.
31	los0_lmt_int_sts	RO	0x0	Ref0 Monitor LOS Threshold Exceeded interrupt status. When set, indicates that the number of times LOSMON0 declared LOS exceeded the programmed limit. This bit reflects the losmon0_los_lmt_evt.
30:28	dpll_lol_lmt_int_sts	RO	0x0	DPLL Loss-of-lock Threshold Exceeded interrupt status. When set, indicates that the number of times the DPLL lost lock exceeded the programmed limit. This bit reflects the dpll_lol_lmt_evt.
27	apll_lol_lmt_int_sts	RO	0x0	APLL Loss-of-lock Threshold Exceeded interrupt status. When set, indicates that the number of times the APLL lost lock exceeded the programmed limit. This bit reflects the apll_lol_lmt_evt.
26	freq3_update_int_sts	RO	0x0	Ref3 Frequency Monitor offset valid interrupt status. When set, indicates the FREQMON3 asserted freq_update at some point. This bit reflects the freqmon3_freq_update_evt bit.
25	freq2_update_int_sts	RO	0x0	Ref2 Frequency Monitor offset valid interrupt status. When set, indicates the FREQMON2 asserted freq_update at some point. This bit reflects the freqmon2_freq_update_evt bit.
24	freq1_update_int_sts	RO	0x0	Ref1 Frequency Monitor offset valid interrupt status. When set, indicates the FREQMON1 asserted freq_update at some point. This bit reflects the freqmon1_freq_update_evt bit.
23	freq0_update_int_sts	RO	0x0	Ref0 Frequency Monitor offset valid interrupt status. When set, indicates the FREQMON0 asserted freq_update at some point. This bit reflects the freqmon0_freq_update_evt bit.
22	freq3_int_sts	RO	0x0	Ref3 Frequency Monitor interrupt status. When set, indicates the FREQMON3 declared freq_fail at some point. This bit reflects the freqmon3_freq_fail_evt bit.
21	freq2_int_sts	RO	0x0	Ref2 Frequency Monitor interrupt status. When set, indicates the FREQMON2 declared freq_fail at some point. This bit reflects the freqmon2_freq_fail_evt bit.
20	freq1_int_sts	RO	0x0	Ref1 Frequency Monitor interrupt status. When set, indicates the FREQMON1 declared freq_fail at some point. This bit reflects the freqmon1_freq_fail_evt bit.
19	freq0_int_sts	RO	0x0	Ref0 Frequency Monitor interrupt status. When set, indicates the FREQMON0 declared freq_fail at some point. This bit reflects the freqmon0_freq_fail_evt bit.

INT_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
18	xtal_int_sts	RO	0x0	XTAL Monitor Loss-of-Signal interrupt status. When set, indicates the XTALMON declared LOS at some point. This bit reflects the xtal_los_evt bit.
17	los4_int_sts	RO	0x0	Ref4 Monitor Loss-of-Signal interrupt status. When set, indicates the LOSMON4 declared LOS at some point. This bit reflects the losmon4_los_evt bit.
16	los3_int_sts	RO	0x0	Ref3 Monitor Loss-of-Signal interrupt status. When set, indicates the LOSMON3 declared LOS at some point. This bit reflects the losmon3_los_evt bit.
15	los2_int_sts	RO	0x0	Ref2 Monitor Loss-of-Signal interrupt status. When set, indicates the LOSMON2 declared LOS at some point. This bit reflects the losmon2_los_evt bit.
14	los1_int_sts	RO	0x0	Ref1 Monitor Loss-of-Signal interrupt status. When set, indicates the LOSMON1 declared LOS at some point. This bit reflects the losmon1_los_evt bit.
13	los0_int_sts	RO	0x0	Ref0 Monitor Loss-of-Signal interrupt status. When set, indicates the LOSMON0 declared LOS at some point. This bit reflects the losmon0_los_evt bit.
12:10	dpll_state_ch_int_sts	RO	0x0	DPLL State Change interrupt status. When set, indicates the DPLL changed state at some point. This bit reflects the dpll_state_ch_evt bit.
9:7	dpll_holdover_int_sts	RO	0x0	DPLL Holdover interrupt status. When set, indicates the DPLL entered holdover at some point. This bit reflects the dpll_holdover_evt bit.
6:4	dpll_lol_int_sts	RO	0x0	DPLL Loss-of-lock interrupt status. When set, indicates that the DPLL lost lock at some point. These bits reflect the dpll_lol_evt bit.
3:2	reserved	RO	0x0	Reserved
1	apll_lock_int_sts	RO	0x0	APLL Lock interrupt stats. When set, indicates that the frequency-based lock detector is in lock.
0	apll_lol_int_sts	RO	0x0	APLL Loss-of-lock interrupt status. When set APLL lost lock at some point. This bit reflects the value of apll_ldet_lol_evt bit.

4.3 SSI

Slave Serial Interface Registers.

Table 15. SSI Register Index

Offset (Hex)	**Register Module Base Address: 0x50**	
	Register Name	**Register Description**
0x0	I2C_FLTR_CNFG	I2C Filter
0x1	I2C_TIMING_CNFG	I2C Timing
0x2	I2C_ADDR_CNFG	I2C Address
0x3	SPI_CNFG	SPI Configuration
0x4	SSI_GLOBAL_CNFG	Slave Serial Interface Global Configuration
0x5	SCL_SCLK_PAD_CNFG	SCL_SCLK Pad Configuration
0x6	SDA_SDIO_PAD_CNFG	SDA_SDIO Pad Configuration
0x7	SDO_A1_PAD_CNFG	SDO_A1 Pad Configuration
0x8	NCS_A0_PAD_CNFG	NCS_A0 Pad Configuration
0x9	I2C_EVENT	I2C CRC Error Count
0xA	I2C_CRC_ERR_CNT_EVENT	I2C CRC Error Count

4.3.1 I2C_FLTR_CNFG

I2C Filter.

I2C_FLTR_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	reserved	RO	0x0	Reserved
3:0	i2c_spike_fltr	RW	0x3	I2C digital spike filter duration. Controls the duration of the digital spike filters on the SCL and SDA inputs, specified in number of system clock cycles (16.7 ns). 0 disables filtering.

4.3.2 I2C_TIMING_CNFG

I2C Timing.

I2C_TIMING_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	i2c_sda_high_hold	RW	0x1	I2C transmit one bit delay. Delays transmission of '1' value by this number of 67ns periods (6 system clock cycles).
3:0	i2c_sda_low_hold	RW	0x1	I2C transmit zero bit delay. Delays transmission of '0' value by this number of 67ns periods (6 system clock cycles). Allows data-hold-times on strongly pulled-down '0' value bits to be set to match data-hold-times on weakly (resistively) pulled-up '1' value bits.

4.3.3 I2C_ADDR_CNFG

I2C Address.

I2C_ADDR_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7	reserved	RO	0x0	Reserved
6:0	i2c_addr	RW	0x9	I2C device address. Sets I2C device address that the SSI will acknowledge and accept accesses on. The bottom three bits are selected using the table provided for i2c_addr_sel field.

4.3.4 SPI_CNFG

SPI Configuration.

SPI_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	reserved	RO	0x0	Reserved
3	spi_del_out	RW	0x0	SDO delay. Selects the delay for driving SDO. 0x0 = SDO is driven on opposite SCLK edge than the sampling edge 0x1 = SDO is delayed one half cycle of SCLK
2	reserved	RO	0x0	Reserved
1	spi_clk_sel	RW	0x0	SDI sampling edge selection. Selects the sclk edge for input sampling. 0x0 = SDI is sampled on rising SCLK edge 0x1 = SDI is sampled on falling SCLK edge
0	spi_3wire	RW	0x0	Select SPI 3 or 4-wire mode. 0x0 = Normal 4-wire SPI. Data is received on SDA_SDIO, and transmitted on the SDO. 0x1 = 3-wire SPI. Data is received and transmitted SDA_SDIO

4.3.5 SSI_GLOBAL_CNFG

Slave Serial Interface Global Configuration.

SSI_GLOBAL_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	reserved	RO	0x0	Reserved
3	i2c_crc_en	RW	0x0	I2C CRC Enable. Enables the I2C CRC check. 0x0 = CRC disabled 0x1 = CRC enabled
2	ssi_addr_size	RW	0x1	SSI address size. When '0' the SSI expects 1-byte CSR addresses; when '1' the SSI expects 2-byte CSR addresses. Upper address bits are taken from the SSI's page register to create a full 32-bit CSR address. 0x0 = 1-byte address 0x1 = 2-byte address
1:0	ssi_enable	RW	0x2	SSI mode. Selects the serial interface mode. 0x0 = SSI is disabled 0x1 = SSI is in I2C mode 0x2 = SSI is in SPI mode 0x3 = Reserved

4.3.6 SCL_SCLK_PAD_CNFG

SCL_SCLK Pad Configuration.

SCL_SCLK_PAD_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:3	reserved	RO	0x2	Reserved
2:0	pad_scl_sclk_drv	RW	0x6	Drive strength. Drive Strength 0x0 = Open drain Output mode. Standard mode. 0x1 = Open drain Output mode. Fast mode. 0x2 = Reserved 0x3 = Open drain Output mode. Fast mode plus. 0x4 = Reserved 0x5 = Reserved 0x6 = CMOS Output mode and power supply of 1.8V. 0x7 = Reserved

4.3.7 SDA_SDIO_PAD_CNFG

SDA_SDIO Pad Configuration.

SDA_SDIO_PAD_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:3	reserved	RO	0x2	Reserved
2:0	pad_sda_sdio_drv	RW	0x6	Drive strength. Drive Strength 0x0 = Open drain Output mode. Standard mode. 0x1 = Open drain Output mode. Fast mode. 0x2 = Reserved 0x3 = Open drain Output mode. Fast mode plus. 0x4 = Reserved 0x5 = Reserved 0x6 = CMOS Output mode and power supply of 1.8V. 0x7 = Reserved

4.3.8 SDO_A1_PAD_CNFG

SDO_A1 Pad Configuration.

SDO_A1_PAD_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:3	reserved	RO	0x2	Reserved
2:0	pad_sdo_a1_drv	RW	0x6	Drive strength. Drive Strength 0x0 = Open drain Output mode. Standard mode. 0x1 = Open drain Output mode. Fast mode. 0x2 = Reserved 0x3 = Open drain Output mode. Fast mode plus. 0x4 = Reserved 0x5 = Reserved 0x6 = CMOS Output mode and power supply of 1.8V. 0x7 = Reserved

4.3.9 NCS_A0_PAD_CNFG

NCS_A0 Pad Configuration.

NCS_A0_PAD_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:3	reserved	RO	0x2	Reserved
2:0	pad_ncs_a0_drv	RW	0x6	Drive strength. Drive Strength 0x0 = Open drain Output mode. Standard mode. 0x1 = Open drain Output mode. Fast mode. 0x2 = Reserved 0x3 = Open drain Output mode. Fast mode plus. 0x4 = Reserved 0x5 = Reserved 0x6 = CMOS Output mode and power supply of 1.8V. 0x7 = Reserved

4.3.10 I2C_EVENT

I2C CRC Error Count.

I2C_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	i2c_crc_err_evt	RW1C	0x0	I2C CRC Error Count. This bits indicates a CRC error was detected. It can be cleared by writing a one to it.

4.3.11 I2C_CRC_ERR_CNT_EVENT

I2C CRC Error Count.

I2C_CRC_ERR_CNT_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:0	i2c_crc_err_cnt	RW	0x0	I2C CRC Error Count. This counter increments each time the I2C interface detects a CRC error, and saturates at 0xFF. It is cleared by writing it to 0x0, and may be preset by writing the desired value. Preset may be used either as a debug tool.

4.4 XO

Crystal Oscillator, Input Buffer and Reference Select Registers.

Table 16. XO Register Index

Offset (Hex)	Register Module Base Address: 0x60	
	Register Name	Register Description
0x0	XO_CNFG	XO Buffer Configuration

4.4.1 XO_CNFG

XO Buffer Configuration.

XO_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:16	reserved	RO	0x0	Reserved
15:12	xobuf_digicap_x2	RW	0x0	XOUT internal tunning cap select 0x0 = 5.5pF 0x1 = 5.5pF + 504fF 0x2 = 5.5pF + 1.008pF 0x3 = 5.5pF + 1.512pF 0x4 = 5.5pF + 2.016pF 0x5 = 5.5pF + 2.520pF 0x6 = 5.5pF + 3.024pF 0x7 = 5.5pF + 3.528pF 0x8 = 5.5pF + 4.032pF 0x9 = 5.5pF + 4.536pF
xobuf_digicap_x2 (continued)				0xA = 5.5pF + 5.040pF 0xB = 5.5pF + 5.544pF 0xC = 5.5pF + 6.048pF 0xD = 5.5pF + 6.552pF 0xE = 5.5pF + 7.056pF 0xF = 5.5pF + 7.560pF
11:8	xobuf_digicap_x1	RW	0x0	XIN internal tunning cap select 0x0 = 5.5pF 0x1 = 5.5pF + 504fF 0x2 = 5.5pF + 1.008pF 0x3 = 5.5pF + 1.512pF 0x4 = 5.5pF + 2.016pF 0x5 = 5.5pF + 2.520pF 0x6 = 5.5pF + 3.024pF 0x7 = 5.5pF + 3.528pF 0x8 = 5.5pF + 4.032pF 0x9 = 5.5pF + 4.536pF
xobuf_digicap_x1 (continued)				0xA = 5.5pF + 5.040pF 0xB = 5.5pF + 5.544pF 0xC = 5.5pF + 6.048pF 0xD = 5.5pF + 6.552pF 0xE = 5.5pF + 7.056pF 0xF = 5.5pF + 7.560pF
7:3	reserved	RO	0x0	Reserved
2	xo_fixedcap_on	RW	0x0	XO fixed capacitance enabled. XO fixed capacitance enabled. 0x0 = Fixed capacitance disabled 0x1 = Fixed capacitance enabled
1:0	reserved	RW	0x3	Reserved

4.5 APLL

APLL Registers.

Table 17. APLL Register Index

Offset (Hex)	Register Module Base Address: 0x80	
	Register Name	Register Description
0x0	LPF_TC_CNFG	VCO Temp Compensation Configuration
0x1	VCO_CNFG	VCO Configuration
0x10	APLL_FB_DIV_FRAC_CNFG	APLL Feedback Divider Fraction
0x18	APLL_FB_DIV_INT_CNFG	APLL Feedback Divider Integer
0x1C	APLL_DCD_CAL_CNFG	APLL DCD Calibration Configuration
0x22	APLL_COMBO_OP_CNFG	APLL Combo operators
0x30	APLL_COMBO_OFFSET_CTRL	APLL Combo offset
0x3A	APLL_EVENT	APLL Lock Events
0x3B	APLL_LOL_EVENT	APLL Loss of Lock Configuration
0x3D	APLL_STS	APLL Lock Status

4.5.1 LPF_TC_CNFG

VCO Temp Compensation Configuration.

LPF_TC_CNFG Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
7	reserved	RO	0x0	Reserved
6:4	cnf_lpf_res	RW	0x6	Loop filter resistor setting. No description 0x0 = Rs=4000ohm 0x1 = Rs=2000ohm 0x2 = Rs=1330ohm 0x3 = Rs=1000ohm 0x4 = Rs=800ohm 0x5 = Rs=677ohm 0x6 = Rs=570ohm 0x7 = Rs=500ohm
3:0	reserved	RW	0x8	Reserved

4.5.2 VCO_CNFG

VCO Configuration.

VCO_CNFG Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
7:5	vco_div	RW	0x2	vco divider configuration 0x0 = Reserved 0x1 = Reserved 0x2 = 2 0x3 = 3 0x4 = 4 0x5 = 5 0x6 = Reserved 0x7 = 7
4:0	reserved	RW	0x13	Reserved

4.5.3 APLL_FB_DIV_FRAC_CNFG

APLL Feedback Divider Fraction.

APLL_FB_DIV_FRAC_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:38	reserved	RO	0x0	Reserved
37:0	apll_fb_div_frac	RW	0x3851E B851E	APLL Feedback Divider Fraction. APLL feedback divider numerator value. The denominator is a fixed value of 2^38.

4.5.4 APLL_FB_DIV_INT_CNFG

APLL Feedback Divider Integer.

APLL_FB_DIV_INT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:10	reserved	RO	0x0	Reserved
9:0	apll_fb_div_int	RW	0x7A	APLL Feedback Divider Integer. APLL feedback divider integer value.

4.5.5 APLL_DCD_CAL_CNFG

APLL DCD Calibration Configuration.

APLL_DCD_CAL_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15	reserved	RO	0x0	Reserved
14	integer_mode	RW	0x0	APLL integer mode. Set this bit to 0x1 when the apll_fb_div_frac = 0x0 AND the DPLL is in freerun mode.
13:0	reserved	RO	0x1a24	Reserved

4.5.6 APLL_COMBO_OP_CNFG

APLL Combo operators.

APLL_COMBO_OP_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:11	reserved	RO	0x0	Reserved
10	apll_combo_sum_hold	RW	0x0	Enable or hold the output from the combo summation. Use hold during re-arrangements of the combo setup. 0x0 = The combo sum value is continuously updated and output 0x1 = At the 0-> transition, the combo sum value is stored and held
9:8	ts_delta_freq_op	RW	0x0	This field controls whether Time Sync Delta Freq gets used. 0x0 = Time Sync contribution is ignored (mathematically zeroed) 0x1 = Time Sync contribution is logically removed (mathematical add) 0x2 = Time Sync contribution is logically added (mathematical subtract) 0x3 = Reserved
7:6	dpll0_fb_ffo_op	RW	0x0	This field controls whether DPLL_0 feedback FFO gets used. 0x0 = DPLL0 contribution is ignored (mathematically zeroed) 0x1 = DPLL0 contribution is logically removed (mathematical add) 0x2 = DPLL0 contribution is logically added (mathematical subtract) 0x3 = Reserved

APLL_COMBO_OP_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
5:4	dpll2_delta_freq_op	RW	0x0	This field controls whether the DPLL2 delta_freq gets used. 0x0 = DPLL2 contribution is ignored (mathematically zeroed) 0x1 = DPLL2 contribution is logically added (mathematical add) 0x2 = DPLL2 contribution is logically removed (mathematical subtract) 0x3 = Reserved
3:2	dpll1_delta_freq_op	RW	0x0	This field controls whether the DPLL1 delta_freq gets used. 0x0 = DPLL1 contribution is ignored (mathematically zeroed) 0x1 = DPLL1 contribution is logically added (mathematical add) 0x2 = DPLL1 contribution is logically removed (mathematical subtract) 0x3 = Reserved
1:0	dpll0_delta_freq_op	RW	0x0	This field controls whether the DPLL0 delta_freq gets used. 0x0 = DPLL0 contribution is ignored (mathematically zeroed) 0x1 = DPLL0 contribution is logically added (mathematical add) 0x2 = DPLL0 contribution is logically removed (mathematical subtract) 0x3 = Reserved

4.5.7 APLL_COMBO_OFFSET_CTRL

APLL Combo offset.

APLL_COMBO_OFFSET_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:39	reserved	RO	0x0	Reserved
38:0	apll_combo_offset	RW	0x0	signed offset value added to the sum of all combo inputs for the APLL This is a 2's complement value. The units are 2^-44 * 1e6 [ppm]

4.5.8 APLL_EVENT

APLL Lock Events.

APLL_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:5	reserved	RO	0x0	Reserved
4	apll_lol_lmt_evt	RW1C	0x0	APLL LOL Limit Reached Event. Set while the APLL Loss-of-Lock counter (apll_lol_cnt) exceeds the threshold set in apll_lol_cnt_thresh. This bit cannot be cleared by software while the condition persists (i.e., the apll_lol_cnt should be cleared before this bit can be cleared). 0x0 = Loss-of-lock counter has not exceeded the threshold since the last time the bit was cleared 0x1 = Loss-of-lock counter exceeded the threshold since the last time the bit was cleared
3:1	reserved	RW1C	0x0	Reserved
0	apll_ldet_lol_evt	RW1C	0x0	APLL LOL Event. Set to 1 when the APLL frequency-based lock status transitions from locked to unlocked. Once asserted, this bit will remain asserted until cleared by a write of '1' to this bit position.

4.5.9 APLL_LOL_EVENT

APLL Loss of Lock Configuration.

APLL_LOL_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	reserved	RO	0x0	Reserved
3:0	apll_lol_cnt	RW	0x0	APLL Loss-of-Lock Counter. This counter increments each time the APLL frequency-based lock status deasserts, and saturates at 0xF. It is cleared by writing it to 0x0, and may be preset by writing the desired value.

4.5.10 APLL_STS

APLL Lock Status.

APLL_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	ldet_lock_sts	RO	0x0	APLL lock status. Set to 1 when the frequency-based APLL lock detector reports a lock 0x0 = Unlocked 0x1 = Locked

4.6 OUTBUF

Output Buffer Registers.

Table 18. OUTBUF Register Index

Offset (Hex)	Register Module Base Address: 0x100	
	Register Name	**Register Description**
0x0	OUT_CNFG	Output Buffer General Configuration
0x2	OUT_BIAS_CNFG	Output Driver Bias Cal Value
0x4	OUT_CTRL	Output Pad Enables and Bias Cal Initialization
0x7	OUT_BIAS_CAL_STS	Output Driver Bias Cal Status
0x6	OUT_SPARE	Spare Output Buffer Bits

4.6.1 OUT_CNFG

Output Buffer General Configuration.

OUT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:6	out_boost	RW	0x0	Output Voltage Swing Boost. Value is the percent of increased swing. 0x0 = Boost = 0 0x1 = Boost 15% 0x2 = Boost 15% 0x3 = Boost up to 30%
5	out_pull_down	RW	0x0	Output Buffer Internal Pull-down Enable. when set, 50 Ohm to ground is enabled 0x0 = Pull-down on output is disabled 0x1 = Pull-down on output is enabled

OUT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
4	out_pull_up	RW	0x0	Output Buffer Internal Pull-up Enable. when set, 50 Ohm to power supply is enabled 0x0 = Pull-up on output is disabled 0x1 = Pull-up on output is enabled
3	out_pd	RW	0x1	Output Buffer Power Down. Setting this bit to 0x1 will powered-down and tri-stated the output, it will also disable the corresponding IOD. 0x0 = Output is powered 0x1 = Output is powered-down
2	out_cmos_same_phase	RW	0x0	Output CMOS Same Phase. P and N output pins will be same phase when CMOS mode is enabled. Applies to OUT 8-11 only. 0x0 = P and N outputs are 180 degrees out of phase when CMOS mode is enabled. 0x1 = P and N outputs are same phase when CMOS mode is enabled.
1:0	out_style	RW	0x3	Output Style. Selects Output Style 0x0 = LVDS 0x1 = CMOS 0x2 = CML 0x3 = HCSL

4.6.2 OUT_BIAS_CNFG

Output Driver Bias Cal Value.

OUT_BIAS_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:6	reserved	RO	0x0	Reserved
5:0	out_ibias	RW	0x20	Output Driver Bias Cal Value. Sets output driver bias cal value

4.6.3 OUT_CTRL

Output Pad Enables and Bias Cal Initialization.

OUT_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:2	reserved	RO	0x0	Reserved
1	oe_source_sel	RW	0x1	Output Driver Enable Source. Selects between GPIO and out_driver_en bit control of the synchronous output driver enable. 0x0 = GPIO 0x1 = out_driver_en field
0	out_driver_en	RW	0x0	Output Pad Synchronous Enable. Synchronously enables/disables output drivers when oe_source_sel is 1. This bit also enables/disables the corresponding IOD. Differential clocks will stop P = low, N = high LVCMOS mode with cmos_same_phase = 1 will stop P = low, N = low LVCMOS mode with cmos_same_phase = 1 will stop P = low, N = high 0x0 = Disable 0x1 = Enable

4.6.4 OUT_BIAS_CAL_STS

Output Driver Bias Cal Status.

OUT_BIAS_CAL_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:6	reserved	RO	0x0	Reserved
5:0	out_cal_ibias_sts	RO	0x0	Output driver bias cal status. Result of output driver bias calibration

4.6.5 OUT_SPARE

Spare Output Buffer Bits.

OUT_SPARE Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7	out_ntristate	RW	0x0	Output Tristate Enable. Controls tristate/active status of output driver 0x0 = Output driver is in tristate 0x1 = Output driver is active if not powered down using out_pd
6	out_delay	RW	0x0	Output Delay Control. Controls delay at output if enabled by iod_out_delay_en 0x0 = 50ps delay 0x1 = 100ps delay
5:0	out_spare	RW	0x0	

4.7 SYSDIV

System Clock Divider Registers.

Table 19. SYSDIV Register Index

Offset (Hex)	**Register Module Base Address: 0x160**	
	Register Name	**Register Description**
0x0	SYS_DIV_INT_CNFG	System Clock Divider Configuration
0x1	COUNTER_1US_CNFG	System 1 us Counter Configuration
0x2	SYS_DIV_EN_CTRL	System Clock Divider Enable

4.7.1 SYS_DIV_INT_CNFG

System Clock Divider Configuration.

SYS_DIV_INT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:5	reserved	RO	0x0	Reserved
4:0	sys_div_int	RW	0xB	Quadruple System Clock Divide Ratio. The quadruple system clock divide integer value must be set to produce a frequency between 180MHz and 280MHz, divided down from the APLL VCO frequency divided by 4. This clock is divided by 2 to generate the double system clock, and further divided by 2 to generate the system clock. The frequency picked will have side effects on various calculations done in other blocks (LOSMON, DPLL, OTP). Normally expected to be between 210MHz and 240MHz, giving a system clock frequency between 52MHz and 60MHz. The minimum valid value for this field is 8. Note: The count_1us register field must be programmed according to the system clock frequency.

4.7.2 COUNTER_1US_CNFG

System 1 us Counter Configuration.

COUNTER_1US_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:0	count_1us	RW	0x3F	One microsecond counter interval. In system clock cycles, minus 1. This configures counters in the OTP and DPLL. Note: This field must be set appropriately to guarantee a minimum duration of 1us. The typical calculation is: ceiling(Fsysclk) - 1, where Fsysclk is the system clock frequency in MHz.

4.7.3 SYS_DIV_EN_CTRL

System Clock Divider Enable.

SYS_DIV_EN_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	sys_div_en	RW	0x1	System Clock Divider enable 0x1 = Enable 0x0 = Disable

4.8 IOD

Integer Output Divider Registers.

Table 20. IOD Register Index

Offset (Hex)	Register Module Base Address: 0x170	
	Register Name	**Register Description**
0x0	IOD_DIV_CNFG	Integer Output Divider Ratio Configuration Register
0x4	IOD_PHASE_CNFG	Integer Output Divider Phase Adjustment Configuration Register
0x6	IOD_CNFG	Integer Output Divider Configuration Register
0x8	IOD_PHASE_EN_CTRL	Integer Output Divider Phase Adjust Trigger

4.8.1 IOD_DIV_CNFG

Integer Output Divider Ratio Configuration Register.

IOD_DIV_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:23	reserved	RO	0x0	Reserved
22:0	iod_divider	RW	0x28	Integer Output Divider Ratio. Integer output divider ratio.

4.8.2 IOD_PHASE_CNFG

Integer Output Divider Phase Adjustment Configuration Register.

IOD_PHASE_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:0	iod_phase_config	RW	0x0	Integer Output Divider Phase Adjustment Value. This value indicates the number of steps by which the phase of the divider output clock should be adjusted by. The step size is defined by the selected input clock to the IOD (if selecting the VCO clock, the step size is vco_div times the VCO clock period, if selecting an FOD clock, the step size is the FOD clock period). This number is signed. A positive number indicates the edge will be delayed, a negative number indicates the edge will be advanced. This phase is applied when the iod_ph_adj_now is written to 1, or is applied immediately after a divider resync event if the iod_ph_adj_post_sync is set to one.

4.8.3 IOD_CNFG

Integer Output Divider Configuration Register.

IOD_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:6	reserved	RO	0x0	Reserved
5	iod_out_delay_en	RW	0x0	Additional delay enable. Additional delay added to output when selected
4	iod_ph_adj_post_sync	RW	0x1	Phase adjust after SYNC. This bit indicates whether the phase adjust should get applied after a divider sync event or not. 0x0 = Do not apply after a sync event 0x1 = Apply immediately after a sync event
3	reserved	RO	0x0	Reserved
2:0	iod_mux_sel	RW	0x0	Integer Output Divider Clock Select. This register field applies to IOD[11:6] only. 0x0 = VCO/N 0x1 = FOD0 0x2 = FOD1 0x3 = FOD2 0x4 = Time sync 0x5 = Time clock 0x6 = Reserved 0x7 = Reserved

4.8.4 IOD_PHASE_EN_CTRL

Integer Output Divider Phase Adjust Trigger.

IOD_PHASE_EN_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	iod_ph_adj_now	RW1S	0x0	IOD Phase Adjust Trigger. When this bit gets written to one, the phase adjustment programmed in iod_phase_config gets applied. This bit will remain high until the phase adjustment has been fully applied. This bit should not be set to one if the IOD is disabled (if the corresponding out_driver_en bit is low).

4.9 SYSREF

SYSREF Registers.

Table 21. SYSREF Register Index

Offset (Hex)	Register Module Base Address: 0x230	
	Register Name	Register Description
0x0	SYSREF_CNFG	SYSREF Configuration Registers
0x8	SYSREF_CTRL	SYSREF Control Register

4.9.1 SYSREF_CNFG

SYSREF Configuration Registers.

SYSREF_CNFG Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
63:48	reserved	RO	0x0	Reserved
47:40	sysref_pause_count	RW	0x0	SYSREF Pause Count. Number of pauses in a continuous SYSREF pulse/pause event. The valid range is 1 to 255. A value of 0 is invalid.
39:32	sysref_pulse_count	RW	0x1	SYSREF Pulse Count. Number of pulses in a counted SYSREF event or a continuous SYSREF pulse/pause event. The valid range is 1 to 255. A value of 0 is treated as a value of 1.
31:22	reserved	RO	0x0	Reserved

SYSREF_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
21:20	sysref_sro	RW	0x2	SYSREF mode select. This field combined with sysref_srg selects the SYSREF mode. The SYSREF mode numbers are the decimal values of the binary concatenations of sysref_srg & sysref_sro, e.g., SYSREF mode 1 results when sysref_srg = 0x0, and sysref_sro = 0x1. SYSREF mode descriptions. SYSREF mode 0 – Counted SYSREF. Start-trigger is low-to-high for init_sysref; stop-trigger is N/A. SYSREF mode 1 – Continuous SYSREF pulse/pause. Start-trigger is low-to-high for init_sysref; stop-trigger is high-to-low for init_sysref. SYSREF mode 2 – Continuous SYSREF. Start-trigger is low-to-high for init_sysref; stop-trigger is high-to-low for init_sysref. SYSREF mode 3 – Continuous SYSREF. Start-trigger is first rising edge on GPIO[1]]; stop-trigger is low-to-high for clear_sequential. SYSREF mode 4 – Counted SYSREF. Start-trigger is first rising-edge on GPIO[1]; stop-trigger is N/A. SYSREF mode 5 – Counted SYSREF. Start-trigger is first falling-edge on GPIO[1]; stop-trigger is N/A. SYSREF mode 6 – Continuous SYSREF. Start-trigger is rising-edge on GPIO[1]; stop-trigger is falling-edge on GPIO[1]. SYSREF mode 7 – Continuous SYSREF. Start-trigger is falling-edge on GPIO[1]; stop-trigger is rising-edge on GPIO[1]. For SYSREF modes 3, 4, 5, 6 and 7, the trigger source (GPIO[1]) can be overridden by the sysref_tod_trigger_sel bit. SYSREF Event Types: Counted SYSREF. When a start-trigger is detected, the device outputs SYSREF pulses until the number of SYSREF pulses given by sysref_pulse_count is reached then it stops outputting SYSREF pulses. Continuous SYSREF pulse/pause. When a start-trigger is detected, the device outputs a SYSREF pattern consisting of SYSREF pulses followed by pauses (non-pulses). The number of pulses is given by sysref_pulse_count, the number of pauses is given by sysref_pause_count. The pattern repeats until a stop-trigger is detected. Continuous SYSREF. When a start-trigger is detected, the device outputs SYSREF pulses until a stop-trigger is detected. ---------- 0x0 = SYSREF mode 0 or 4, depending on sysref_srg. 0x1 = SYSREF mode 1 or 5, depending on sysref_srg. 0x2 = SYSREF mode 2 or 6, depending on sysref_srg. 0x3 = SYSREF mode 3 or 7, depending on sysref_srg.
19	sysref_trig_from_prime	RW	0x0	SYSREF Output Start Select. This bit determines which rising edge of the internal SYSREF clock, after a SYSREF start trigger is detected, is the timing reference for the start of SYSREF output. 0x0 = Second rising-edge of the internal SYSREF clock 0x1 = First rising-edge of the internal SYSREF clock
18	sysref_srg	RW	0x0	SYSREF mode select. This field combined with sysref_sro selects the SYSREF mode. The SYSREF mode numbers are the decimal values of the binary concatenations of sysref_srg & sysref_sro, e.g., SYSREF mode 1 results when sysref_srg = 0x0, and sysref_sro = 0x1. See the description of sysref_sro for SYSREF mode descriptions. 0x0 = SYSREF modes 0 to 3, depending on sysref_sro 0x1 = SYSREF modes 4 to 7, depending on sysref_sro

SYSREF_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
17	sysref_tod_trigger_sel	RW	0x0	SYSREF TOD Compare Source Select. This bit determines if the tod_compare_evt bit is used as the SYSREF trigger source, instead of GPIO[1]. This bit affects only SYSREF modes 3, 4, 5, 6 and 7. The SYSREF modes are determined by the sysref_srg and sysref_sro fields. 0x0 = SYSREF trigger source is GPIO[1] 0x1 = SYSRE trigger source is tod_compare_evt
16	sysref_is_prime	RW	0x0	SYSREF Trigger Output for Secondary Devices. This field determines if the SYSREF controller outputs SYSREF triggers via GPIO[0] for secondary devices in a SYSREF group. For SYSREF modes 0, 4 and 5 (as per sysref_srg and sysref_sro), the SYSREF controller outputs SYSREF triggers via GPIO[0] as follows: The GPIO[0] output level is held low until a SYSREF start trigger is detected, then on the next rising edge of the internal SYSREF clock, the GPIO[0] output level is changed to high. Then, when the SYSREF count is reached, one and a half SYSREF periods later, the GPIO[0] output level is changed to low. For SYSREF modes 1, 2, 6 and 7, the SYSREF controller outputs SYSREF triggers via GPIO[0] as follows: The GPIO[0] output level is held low until a SYSREF start trigger is detected, then on the next rising edge of the internal SYSREF clock, the GPIO[0] output level is changed to high. Then, when a SYSREF stop trigger is detected, the GPIO[0] output level is changed to low. For SYSREF mode 3, the SYSREF controller does not output SYSREF triggers via GPIO[0], regardless of sysref_is_prime. 0x0 = The SYSREF controller does not output SYSREF triggers via GPIO[0] 0x1 = The SYSREF controller outputs SYSREF triggers via GPIO[0]
15:12	reserved	RO	0x0	Reserved
11:0	out_sysref_sel	RW	0x0	SYSREF Enable. When set, this field configures the output as SYSREF. Bit<0> corresponds to OUT0, bit<1> corresponds to OUT1, etc. 0x0 = Output is a device clock 0x1 = Output is a SYSREF

4.9.2 SYSREF_CTRL

SYSREF Control Register.

SYSREF_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1	sysref_clear_sequential	RW	0x0	SYSREF Stop. This bit is used to stop a SYSREF event for a device in SYSREF mode 3 (as per sysref_srg and sysref_sro). When written to 0x1, this bit causes SYSREF output to stop. This bit needs to be manually written back to 0x0. 0x0 = No action 0x1 = Stops SYSREF event
0	init_sysref	RW	0x0	Manual SYSREF Trigger. For SYSREF modes 0, 1 and 2 (as per sysref_srg and sysref_sro), writing 0x1 to this bit will initiate a SYSREF event. For SYSREF mode 0 this bit is cleared automatically when the SYSREF event completes. For SYSREF modes 1 and 2, a SYSREF event can be stopped by writing 0x0 to this bit. For SYSREF modes 3, 4, 5, 6,and 7, this bit has no effect.

4.10 GPIO

General Purpose IO Registers.

Instances 0-7 apply to GPIO0 to GPIO9.

Instance 8 applies to LOCK.

Table 22. GPIO Register Index

Offset (Hex)	Register Module Base Address: 0x240	
	Register Name	Register Description
0x0	GPIO_CNFG	GPIO Mode Configuration
0x2	GPIO_DEGLITCH_CNFG	GPIO Deglitcher Configuration
0x3	GPIO_PAD_CNFG	GPIO Pad Configuration
0x4	GPIO_STS	GPIO Status

4.10.1 GPIO_CNFG

GPIO Mode Configuration.

GPIO_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15	reserved	RO	0x0	Reserved
14	gpio_deglitch_bypass	RW	0x0	GPIO deglitcher bypass. Enables use of the asynchronous gpio for different functions. When set, will send out a synchronized version of the input and may have a sampling variation of 1 to 2 system clocks 0x0 = Deglitcher in use 0x1 = Deglitcher bypassed
13	gpio_resync	RW	0x0	GPIO resynchronize enable. When the GPIO is configured as an output, setting this bit will cause the internal signal to be resynchronized to the system clock domain before being sent out to the GPIO pin. When the GPIO is configured as an input, setting this bit will cause the input value to be resynchronized to the system clock domain before getting sent to the gpio_sts CSR field. 0x0 = No synchronization 0x1 = Synchronization to the system clock is enabled
12	gpio_pol	RW	0x0,0x1, 0x0,0x0, 0x0	GPIO polarity. inverts input or output 0x0 = Do not invert 0x1 = Invert
11:8	gpio_startup_mode	RW	0xB	GPIO startup function. chooses startup function 0x0 = config_index[0] 0x1 = config_index[1] 0x2 = I2C_addr[0] 0x3 = I2C_addr[1] 0x4 = I2C_addr[2] 0x5 = EEPROM dev_addr[0] 0x6 = EEPROM dev_addr[1] 0x7 = EEPROM dev_addr[2] 0x8 = Both I2C_addr[0] and EEPROM dev_addr[0] 0x9 = Both I2C_addr[1] and EEPROM dev_addr[1]
gpio_startup_mode (continued)				0xA = Both I2C_addr[2] and EEPROM dev_addr[2] 0xB = Pin is unused for startup function selection

GPIO_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:0	gpio_func	RW	0x20,0x1B,0x20,0x20,0x20	GPIO function. Select the function of the corresponding GPIO. For the output functions, the corresponding pad_gpio_oe_b bit must be set to 0. 0x0 = Logic Low (output) 0x1 = Logic High (output) 0x2 = XIN LOS (output) 0x3 = REFIN0 LOS (loss of signal) (output) 0x4 = REFIN1 LOS (loss of signal) (output) 0x5 = REFIN2 LOS (loss of signal) (output) 0x6 = REFIN3 LOS (loss of signal) (output) 0x7 = REFIN4 LOS (loss of signal) (output) 0x8 = REFIN0 LOF (loss of frequency) (output) 0x9 = REFIN1 LOF (loss of frequency) (output)
gpio_func (continued)				0xA = REFIN2 LOF (loss of frequency) (output) 0xB = REFIN3 LOF (loss of frequency) (output) 0xC = REFIN0 ref_qual_sts (output) 0xD = REFIN1 ref_qual_sts (output) 0xE = REFIN2 ref_qual_sts (output) 0xF = REFIN3 ref_qual_sts (output) 0x10 = REFIN4 ref_qual_sts (output) 0x11 = XIN LOS limit (output) 0x12 = REFIN0 LOS limit (output) 0x13 = REFIN1 LOS limit (output)
gpio_func (continued)				0x14 = REFIN2 LOS limit (output) 0x15 = REFIN3 LOS limit (output) 0x16 = REFIN4 LOS limit (output) 0x17 = Load complete (OTP or EEPROM loaded) (output) 0x18 = Device Ready (startup sequence complete) (output) 0x19 = APLL rail low (output) 0x1A = Lock Detector FCL Enable 0x1B = APLL lock (from frequency-based lock detect) (output) 0x1C = APLL lol (output) 0x1D = APLL lol_lmt (output)
gpio_func (continued)				0x1E = APLL LF lock (VCTRL-based lock) (output) 0x1F = FCL lock (aux loop status) (output) 0x20 = General purpose input (input) 0x21 = EXT_SYS_IN (Applies to GPIO1 only. Must set oe_b = 1) 0x22 = EXT_SYS_OUT (Applies to GPIO0 only. Must set oe_be=1 and sysref_is_prime = 1) 0x2A = DPLL0 lock (output) 0x2B = DPLL0 lol (output) 0x2C = DPLL0 lol_lmt (output) 0x2D = DPLL0 state[0] (output) 0x2E = DPLL0 state[1] (output)
gpio_func (continued)				0x2F = DPLL0 state[2] (output) 0x30 = DPLL1 lock (output) 0x31 = DPLL1 lol (output) 0x32 = DPLL1 lol_lmt (output) 0x33 = DPLL1 state[0] (output) 0x34 = DPLL1 state[1] (output) 0x35 = DPLL1 state[2] (output) 0x36 = DPLL2 lock (output) 0x37 = DPLL2 lol (output) 0x38 = DPLL2 lol_lmt (output)

GPIO_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
	gpio_func (continued)			0x39 = DPLL2 state[0] (output) 0x3A = DPLL2 state[1] (output) 0x3B = DPLL2 state[2] (output) 0x3C = TOD compare (output) (set gpio_resync to 0) 0x4C = Global OE (input) 0x4D = Group OE[0:3] (input) 0x4E = Group OE[4:7] (input) 0x4F = Group OE[8:11] (input) 0x50 = OE[0] (input) 0x51 = OE[1] (input)
	gpio_func (continued)			0x52 = OE[2] (input) 0x53 = OE[3] (input) 0x54 = OE[4] (input) 0x55 = OE[5] (input) 0x56 = OE[6] (input) 0x57 = OE[7] (input) 0x58 = OE[8] (input) 0x59 = OE[9] (input) 0x5A = OE[10] (input) 0x5B = OE[11] (input)
	gpio_func (continued)			0x5C = DPLL0_force_freerun (input) 0x5D = DPLL0_force_holdover (input) 0x5E = DPLL0_force_normal (input) 0x5F = DPLL0_force_acquire (input) 0x60 = DPLL0 PBO clear (input) 0x61 = DPLL0 bandwidth selection (input) 0x62 = DPLL0 REFIN_SEL[0] (input) 0x63 = DPLL0 REFIN_SEL[1] (input) 0x64 = DPLL1_force_freerun (input) 0x65 = DPLL1_force_holdover (input)
	gpio_func (continued)			0x66 = DPLL1_force_normal (input) 0x67 = DPLL1_force_acquire (input) 0x68 = DPLL1 PBO clear (input) 0x69 = DPLL1 bandwidth selection (input) 0x6A = DPLL1 REFIN_SEL[0] (input) 0x6B = DPLL1 REFIN_SEL[1] (input) 0x6C = DPLL2_force_freerun (input) 0x6D = DPLL2_force_holdover (input) 0x6E = DPLL2_force_normal (input) 0x6F = DPLL2_force_acquire (input)
	gpio_func (continued)			0x70 = DPLL2 PBO clear (input) 0x71 = DPLL2 bandwidth selection (input) 0x72 = DPLL2 REFIN_SEL[0] (input) 0x73 = DPLL2 REFIN_SEL[1] (input) 0x74 = REFIN0_FORCE_LOS (input) 0x75 = REFIN1_FORCE_LOS (input) 0x76 = REFIN2_FORCE_LOS (input) 0x77 = REFIN3_FORCE_LOS (input) 0x7F = Device Global Interrupt (output) 0x80 = Device interrupt[0] - APLL LOL (frequency-based lock detector) (output)

GPIO_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
	gpio_func (continued)			0x81 = Device interrupt[1] - APLL Lock (frequency-based lock detector) (output) 0x82 = Device interrupt[2] - APLL LF valid (VCTRL-based lock) (output) 0x83 = Device interrupt[3] - APLL FCL lock (aux loop status) (output) 0x84 = Device interrupt[4] - DPLL0 LOL (output) 0x85 = Device interrupt[5] - DPLL1 LOL (output) 0x86 = Device interrupt[6] - DPLL2 LOL (output) 0x87 = Device interrupt[7] - DPLL0 HOLDOVER (output) 0x88 = Device interrupt[8] - DPLL1 HOLDOVER (output) 0x89 = Device interrupt[9] - DPLL2 HOLDOVER (output) 0x8A = Device interrupt[10] - DPLL0 STATE CHANGE (output)
	gpio_func (continued)			0x8B = Device interrupt[11] - DPLL1 STATE CHANGE (output) 0x8C = Device interrupt[12] - DPLL2 STATE CHANGE (output) 0x8D = Device interrupt[13] - REFIN0 LOS (output) 0x8E = Device interrupt[14] - REFIN1 LOS (output) 0x8F = Device interrupt[15] - REFIN2 LOS (output) 0x90 = Device interrupt[16] - REFIN3 LOS (output) 0x91 = Device interrupt[17] - XIN0 LOS (output) 0x92 = Device interrupt[18] - REFIN4 LOS (output) 0x93 = Device interrupt[19] - REFIN0 LOF (output) 0x94 = Device interrupt[20] - REFIN1 LOF (output)
	gpio_func (continued)			0x95 = Device interrupt[21] - REFIN2 LOF (output) 0x96 = Device interrupt[22] - REFIN3 LOF (output) 0x97 = Device interrupt[23] - REFIN0 FREQ UPDATE (output) 0x98 = Device interrupt[24] - REFIN1 FREQ UPDATE (output) 0x99 = Device interrupt[25] - REFIN2 FREQ UPDATE (output) 0x9A = Device interrupt[26] - REFIN3 FREQ UPDATE (output) 0x9B = Device interrupt[27] - APLL LOL LMT (output) 0x9C = Device interrupt[28] - DPLL0 LOL LMT (output) 0x9D = Device interrupt[29] - DPLL1 LOL LMT (output) 0x9E = Device interrupt[30] - DPLL2 LOL LMT (output)
	gpio_func (continued)			0x9F = Device interrupt[31] - REFIN0 LOS LMT (output) 0xA0 = Device interrupt[32] - REFIN1 LOS LMT (output) 0xA1 = Device interrupt[33] - REFIN2 LOS LMT (output) 0xA2 = Device interrupt[34] - REFIN3 LOS LMT (output) 0xA3 = Device interrupt[35] - XIN0 LOS LMT (output) 0xA4 = Device interrupt[36] - REFIN4 LOS LMT (output) 0xA5 = Device interrupt[37] - OTP MANUAL RDY (output) 0xA6 = Device interrupt[38] - OTP/EEPROM CRC ERR (output) 0xA7 = Device interrupt[39] - OTP/EEPROM FAIL EVT (output) 0xA8 = Device interrupt[44] - TIME SYNC TOD COMPARE (output) (set gpio_resync to 0)
	gpio_func (continued)			0xA9 = Device interrupt[45] - TIME SYNC TDC FIFO CNT ALARM (output) 0xAA = Device interrupt[46] - TIME SYNC TDC FIFO OVERRUN (output) 0xAB = Device interrupt[40] - DPLL0 bandwidth selection (output) 0xAC = Device interrupt[41] - DPLL1 bandwidth selection (output) 0xAD = Device interrupt[42] - DPLL2 bandwidth selection (output) 0xAE = Device interrupt[43] - I2C CRC ERR (output) 0xD7 = tdc_clk[0] (output) (set gpio_resync to 0)
	gpio_func (continued)			0xD8 = tdc_clk[1] (output) (set gpio_resync to 0) 0xD9 = tdc_clk[2] (output) (set gpio_resync to 0) 0xDA = tdc_clk[3] (output) (set gpio_resync to 0)

4.10.2 GPIO_DEGLITCH_CNFG

GPIO Deglitcher Configuration.

GPIO_DEGLITCH_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:0	gpio_deglitch_limit	RW	0x0	GPIO deglitcher limit. Sets the limit of spike filter in steps of 1us. Make sure to program APLL.COUNTER_1US_CNFG.count_1us based on the actual system clock frequency for better accuracy. A limit of 0 with gpio_deglitch_bypass=0 will send out a synchronized version of the input and may have a sampling variation of 1 to 2 system clocks

4.10.3 GPIO_PAD_CNFG

GPIO Pad Configuration.

GPIO_PAD_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:6	reserved	RW	0x0	Reserved
5	pad_gpio_rd_b	RW	0x1	Pull down enable. Pull down enable (active low) 0x0 = Enable 0x1 = Disable
4	pad_gpio_ru_b	RW	0x0	Pull up enable. Pull up enable (active low) 0x0 = Enable 0x1 = Disable
3	pad_gpio_oe_b	RW	0x1,0x0, 0x1,0x1, 0x1	Output enable. Enable output buffer (active low). The output buffer must be disabled when gpio_func is configured for an input function. 0x0 = Enable 0x1 = Disable
2:0	pad_gpio_drv	RW	0x6,0x1, 0x6,0x6, 0x6	Drive strength. Drive Strength 0x0 = Open drain Output mode. Standard mode. 0x1 = Open drain Output mode. Fast mode. 0x2 = Open drain Output mode. Not used 0x3 = Open drain Output mode. Fast mode plus. 0x4 = Reserved 0x5 = Reserved 0x6 = CMOS Output mode and power supply of 1.8V. 0x7 = Reserved

4.10.4 GPIO_STS

GPIO Status.

GPIO_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	gpio_sts	RO	0x0	GPIO Status. Indicates status of the GPIO pins. This value can be read as the raw GPIO value or can be synchronized (controlled by the gpio_resync field) Polarity (gpio_pol) is not taken into consideration and will always report the real GPIO state.

4.11 FOD

Fractional Output Divider Registers.

Table 23. FOD Register Index

Offset (Hex)	Register Module Base Address: 0x300	
	Register Name	Register Description
0x0	FOD_CNFG	FOD General Configuration
0x2	FOD_PHASE_CNFG	FOD Write Phase Control
0x8	FOD_NUM_CNFG	FOD Divider Numerator Configuration
0x10	FOD_DEN_CNFG	FOD Divider Denominator Configuration
0x18	FOD_DIV_CNFG	FOD Divider Integer and Fractional Configuration
0x20	FOD_COMBO_OFFSET_CTRL	FOD Combo offset
0x30	FOD_EN_CTRL	FOD enable / power down
0x32	FOD_PHASE_EN_CTRL	FOD Phase Adjust Trigger
0x28	FOD_WRITE_FREQ_CTRL	FOD Write Frequency Control

4.11.1 FOD_CNFG

FOD General Configuration.

FOD_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15	fod_combo_sum_hold	RW	0x0	enable or hold the output from the combo summation. Use hold during re-arrangements of the combo setup 0x0 = The combo sum value is continuously updated and output 0x1 = At the 0-> transition, the combo sum value is stored and held
14:13	fod_dpll0_fb_ffo_op	RW	0x0	This field controls whether DPLL_0 feedback FFO gets used. 0x0 = DPLL0 contribution is ignored (mathematically zeroed) 0x1 = DPLL0 contribution is logically added (mathematical add) 0x2 = DPLL0 contribution is logically removed (mathematical subtract) 0x3 = Reserved
12:11	fod_dpll2_delta_freq_op	RW	0x0	This field controls whether the DPLL2 delta_freq gets used 0x0 = DPLL2 contribution is ignored (mathematically zeroed) 0x1 = DPLL2 contribution is logically added (mathematical add) 0x2 = DPLL2 contribution is logically removed (mathematical subtract) 0x3 = Reserved
10:9	fod_dpll1_delta_freq_op	RW	0x0	This field controls whether the DPLL1 delta_freq gets used 0x0 = DPLL1 contribution is ignored (mathematically zeroed) 0x1 = DPLL1 contribution is logically added (mathematical add) 0x2 = DPLL1 contribution is logically removed (mathematical subtract) 0x3 = Reserved
8:7	fod_dpll0_delta_freq_op	RW	0x0	This field controls whether the DPLL0 delta_freq gets used 0x0 = DPLL0 contribution is ignored (mathematically zeroed) 0x1 = DPLL0 contribution is logically added (mathematical add) 0x2 = DPLL0 contribution is logically removed (mathematical subtract) 0x3 = Reserved
6:5	fod_ts_delta_freq_op	RW	0x0	This fields controls whether the Time Sync delta_freq to FOD gets ignored or added. 0x0 = Time sync channel contribution is ignored (mathematically zeroed) 0x1 = Time sync channel contribution is logically added (mathematical add) 0x2 = Time sync channel contribution is logically removed (mathematical subtract) 0x3 = Reserved

FOD_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
4	fod_ph_adj_post_sync	RW	0x1	Phase adjust after SYNC. This bit indicates whether the phase adjust should get applied after a divider sync event or not. 0x0 = Sync not applied 0x1 = Sync is applied
3:2	reserved	RO	0x1	Reserved
1	fod_sync_mode	RW	0x0	FOD synchronous mode enable. Enables the synchronous mode of the FOD See fod_div_integer for the formulas 0x0 = Non-synchronous mode, fod_div_numerator and fod_div_denominator are not used. 0x1 = Synchronous mode, fod_div_numerator and fod_div_denominator are used.
0	reserved	RW	0x0	Reserved

4.11.2 FOD_PHASE_CNFG

FOD Write Phase Control.

FOD_PHASE_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:10	reserved	RO	0x0	Reserved
9:0	fod_phase_config	RW	0x0	FOD output clock phase adjust. Signed phase change in steps of 1/4 VCO clock cycle. In integer_mode only whole VCO clock cycle adjustments are possible (fod_phase_adj = 4, 8, 12 etc). This phase gets applied when the fod_ph_adj_now gets written to 1, or gets applied immediately after a divider resync event if the fod_ph_adj_post_sync is set to one.

4.11.3 FOD_NUM_CNFG

FOD Divider Numerator Configuration.

FOD_NUM_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:40	reserved	RO	0x0	Reserved
39:0	fod_div_numerator	RW	0x0	FOD divide numerator. FOD divide numerator for Synchronous mode (fod_sync_mode=1) only see fod_div_integer value decode for formula

4.11.4 FOD_DEN_CNFG

FOD Divider Denominator Configuration.

FOD_DEN_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:40	reserved	RO	0x0	Reserved
39:0	fod_div_denominator	RW	0x8000000000	FOD denominator. FOD denominator for Synchronous mode (fod_sync_mode=1) only. see fod_div_integer value decode for formula

4.11.5 FOD_DIV_CNFG

FOD Divider Integer and Fractional Configuration.

FOD_DIV_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:49	reserved	RO	0x0	Reserved
48:40	fod_div_integer	RW	0x14	FOD divide integer part. Integer portion of FOD divide ratio, min value is 4, max value is 510, programmed value gets split between the hi and lo pulse widths. The lo pulse width will be +1 larger for odd values. All the divider values to the FOD (fod_div_numerator, fod_div_denominator, fod_div_fraction and fod_div_integer) gets updated when this field gets written. if fod_sync_mode=0: FOD divide ratio = fod_div_integer + fod_div_fraction[39:0]/2^40 if fod_sync_mode=1: FOD divide ratio = fod_div_integer + (fod_div_fraction[15:0] + fod_div_numerator/fod_div_denominator))/2^16
39:0	fod_div_fraction	RW	0x0	FOD divide fractional part. In Synthesizer & DCO mode, all bits are used. In Synchronous mode, only bits [15:0] are used, bits [39:16 are ignored. see fod_div_integer value decode for formula

4.11.6 FOD_COMBO_OFFSET_CTRL

FOD Combo offset.

FOD_COMBO_OFFSET_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:39	reserved	RO	0x0	Reserved
38:0	fod_combo_offset	RW	0x0	signed offset value added to the sum of all combo inputs for the FOD. This is a 2's complement value. The units are 2^-44 * 1e6 [ppm]

4.11.7 FOD_EN_CTRL

FOD enable / power down.

FOD_EN_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	fod_enable	RW	0x0	FOD Enable. When set, enables the FOD. This bit also acts as an active low reset of the FOD 0x0 = Disable & reset FOD 0x1 = Enable

4.11.8 FOD_PHASE_EN_CTRL

FOD Phase Adjust Trigger.

FOD_PHASE_EN_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	fod_ph_adj_now	RW	0x0	FOD Phase Adjust Trigger. When this bit gets written to one, the phase adjustment programmed in fod_phase_config gets applied. This bit self clears to 0.

4.11.9 FOD_WRITE_FREQ_CTRL

FOD Write Frequency Control.

FOD_WRITE_FREQ_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:33	reserved	RO	0x0	Reserved
32:0	fod_write_freq	RW	0x0	DCO mode write_frequency fractional offset. signed fractional frequency offset. Resolution 1LSB = 2^-44, range ±244 ppm

4.12 INPUTBUF

Input Buffer Registers.

Table 24. INPUTBUF Register Index

Offset (Hex)	Register Module Base Address: 0x400	
	Register Name	**Register Description**
0x0	INPUT_CNFG	Input Buffer Configuration

4.12.1 INPUT_CNFG

Input Buffer Configuration.

INPUT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7	ib_pull_down	RW	0x1	Pull down control for CLKINp input 0x0 = Disable. 0x1 = Enable.
6	ib_pull_up	RW	0x1	Pull up control for CLKINn input 0x0 = Disable. 0x1 = Enable.
5	ib_ref_mux_sel_gating_byp	RW	0x0	Bypass fanout gating for input reference clocks which are automatically gated by ref*_mux_sel. Used for hitless switching 0x0 = Clock is gated 0x1 = Clock gating is bypassed
4	ib_cmos_sel	RW	0x0	Clock input pad CMOS/Differential selection. Clock input pad CMOS/Differential selection. 0x0 = Differential input is selected 0x1 = CMOS input is selected
3	ib_p_n_diff_sel	RW	0x1	Clock input pad PMOS/NMOS selection. Clock input pad PMOS/NMOS selection according to the common mode voltage of the provided input signal. 0x0 = PMOS input pair is enabled (low common mode voltage) 0x1 = NMOS input pair is enabled (higher common mode voltage)
2	ib_en_ac_couple_bias	RW	0x0	Clock input pad internal DC bias enable. When the differential clock input signal is AC-coupled external to the device, the internal DC bias voltage must be enabled. 0x0 = Internal DC bias is disabled (input signal is DC-coupled) 0x1 = Internal DC bias is enabled (input signal is AC-coupled)

INPUT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
1	ib_en_selfbias_cmos	RW	0x0	Clock input pad internal self-bias enable. When the single-ended reference clock input signal is AC-coupled external to the device, the internal self-bias voltage must be enabled. 0x0 = Internal self-bias is disabled (input signal is DC-coupled) 0x1 = Internal self-bias is enabled (input signal is AC-coupled)
0	ib_en_inbuff	RW	0x0	Clock input pad enable. Clock input pad enable. The clock input pad must be enabled in Jitter Attenuator mode and should be left disabled in synthesizer/DCO mode. 0x0 = Disable 0x1 = Enable

4.13 INPUTMUX

Input Mux Registers.

Table 25. INPUTMUX Register Index

Offset (Hex)	Register Module Base Address: 0x410	
	Register Name	**Register Description**
0x0	REF_SEL_CNFG	Reference Mux Configuration

4.13.1 REF_SEL_CNFG

Reference Mux Configuration.

REF_SEL_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:23	reserved	RO	0x0	Reserved
22:21	dpll_fb_tdc_mux_sel	RW	0x0	DPLL feedback to TDC mux select 0x0 = DPLL0 feedback 0x1 = DPLL1 feedback 0x2 = DPLL2 feedback 0x3 = Reserved
20:18	ref6_mux_sel	RW	0x0	ref6 clock mux select 0x0 = CLKIN0 0x1 = CLKIN1 0x2 = CLKIN2 0x3 = CLKIN3 0x4 = nCLKIN0 0x5 = nCLKIN1 0x6 = nCLKIN2 0x7 = nCLKIN3
17:15	ref5_mux_sel	RW	0x0	ref5 clock mux select 0x0 = CLKIN0 0x1 = CLKIN1 0x2 = CLKIN2 0x3 = CLKIN3 0x4 = nCLKIN0 0x5 = nCLKIN1 0x6 = nCLKIN2 0x7 = nCLKIN3

REF_SEL_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
14:12	ref4_mux_sel	RW	0x0	ref4 clock mux select 0x0 = CLKIN0 0x1 = CLKIN1 0x2 = CLKIN2 0x3 = CLKIN3 0x4 = nCLKIN0 0x5 = nCLKIN1 0x6 = nCLKIN2 0x7 = nCLKIN3
11:9	ref3_mux_sel	RW	0x0	ref3 clock mux select 0x0 = CLKIN0 0x1 = CLKIN1 0x2 = CLKIN2 0x3 = CLKIN3 0x4 = nCLKIN0 0x5 = nCLKIN1 0x6 = nCLKIN2 0x7 = nCLKIN3
8:6	ref2_mux_sel	RW	0x0	ref2 clock mux select 0x0 = CLKIN0 0x1 = CLKIN1 0x2 = CLKIN2 0x3 = CLKIN3 0x4 = nCLKIN0 0x5 = nCLKIN1 0x6 = nCLKIN2 0x7 = nCLKIN3
5:3	ref1_mux_sel	RW	0x0	ref1 clock mux select 0x0 = CLKIN0 0x1 = CLKIN1 0x2 = CLKIN2 0x3 = CLKIN3 0x4 = nCLKIN0 0x5 = nCLKIN1 0x6 = nCLKIN2 0x7 = nCLKIN3
2:0	ref0_mux_sel	RW	0x0	ref0 clock mux select 0x0 = CLKIN0 0x1 = CLKIN1 0x2 = CLKIN2 0x3 = CLKIN3 0x4 = nCLKIN0 0x5 = nCLKIN1 0x6 = nCLKIN2 0x7 = nCLKIN3

4.14 INPUTDIV

Input Buffer Registers.

Table 26. INPUTDIV Register Index

Offset (Hex)	**Register Module Base Address: 0x420**	
	Register Name	**Register Description**
0x0	INPUT_DIV_CNFG	Input Divider Configuration

4.14.1 INPUT_DIV_CNFG

Input Divider Configuration.

INPUT_DIV_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:24	reserved	RO	0x0	Reserved
23	id_enb	RW	0x0	Input Divider Enable. This bit is active low. 0x0 = Enable 0x1 = Disable
22	id_setb	RW	0x0	Input Divider Set. When this bit = 0, the divider is halted (held in set mode) and the divider output will not toggle. 0x0 = Set 0x1 = Not set
21	reserved	RO	0x0	Reserved
20	id_byp_en	RW	0x1	Input Divider Bypass. When this bit is = 1, the divider is bypassed and the divider input clock is passed directly to the DPLL reference input for an effective divide ratio of 1. The frequency of the clock passed to the DPLL reference input must not exceed 33MHz. 0x0 = Divider is not bypassed 0x1 = Divider is bypassed
19:0	id_div_pgm	RW	0x0	Input Divider ratio. The divider input clock frequency is divided by this value. The minimum divide value is 2. The divider may be bypassed for an effective divide ratio of 1. The frequency of the clock passed to the DPLL reference input must not exceed 33MHz. To support hitless switching between reference input clocks, the nominal frequencies of the divided reference input clocks passed to the DPLL reference input must be equal.

4.15 INPUTOFFSET

Reference Offset Registers.

Table 27. INPUTOFFSET Register Index

Offset (Hex)	Register Module Base Address: 0x430	
	Register Name	**Register Description**
0x0	DPLL_PHASE_OFFSET_CNFG	DPLL Phase Offset Configuration

4.15.1 DPLL_PHASE_OFFSET_CNFG

DPLL Phase Offset Configuration.

DPLL_PHASE_OFFSET_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
127:64	reserved	RO	0x0	Reserved
63:48	ref3_phase_offset	RW	0x0	Phase Offset for ref 3. Manually sets the phase offset between the reference and feedback clocks. This is a 16-bit 2's complement value. The resolution is the TDC resolution / 2 (~ 9.2 ps) and the range is ~ +-300 ns. This allows all outputs to be adjusted in terms of their phase relationship to the input. All outputs move together using this precision setting. This field is not used when hitless switching is enabled. This register is atomic. When the most significant byte (bits [15:8]) is written, the new value is applied to the DPLL.
47:32	ref2_phase_offset	RW	0x0	Phase Offset for ref 2. Manually sets the phase offset between the reference and feedback clocks. This is a 16-bit 2's complement value. The resolution is the TDC resolution / 2 (~ 9.2 ps) and the range is ~ +-300 ns. This allows all outputs to be adjusted in terms of their phase relationship to the input. All outputs move together using this precision setting. This field is not used when hitless switching is enabled. This register is atomic. When the most significant byte (bits [15:8]) is written, the new value is applied to the DPLL.

DPLL_PHASE_OFFSET_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:16	ref1_phase_offset	RW	0x0	Phase Offset for ref 1. Manually sets the phase offset between the reference and feedback clocks. This is a 16-bit 2's complement value. The resolution is the TDC resolution / 2 (~ 9.2 ps) and the range is ~ +-300 ns. This allows all outputs to be adjusted in terms of their phase relationship to the input. All outputs move together using this precision setting. This field is not used when hitless switching is enabled. This register is atomic. When the most significant byte (bits [15:8]) is written, the new value is applied to the DPLL.
15:0	ref0_phase_offset	RW	0x0	Phase Offset for ref 0. Manually sets the phase offset between the reference and feedback clocks. This is a 16-bit 2's complement value. The resolution is the TDC resolution / 2 (~ 9.2 ps) and the range is ~ +-300 ns. This allows all outputs to be adjusted in terms of their phase relationship to the input. All outputs move together using this precision setting. This field is not used when hitless switching is enabled. This register is atomic. When the most significant byte (bits [15:8]) is written, the new value is applied to the DPLL.

4.16 TDCAPLL

TDC APLL Registers.

Table 28. TDCAPLL Register Index

Offset (Hex)	Register Module Base Address: 0x440	
	Register Name	**Register Description**
0x2	TDC_FB_DIV_INT_CNFG	TDC Feedback Divider Integer Configuration
0x3	TDC_REF_DIV_CNFG	TDC Reference Divider Configuration
0xA	TDC_CTRL	TDC Filter Status
0xD	TDC_DAC_TRIM_CAL_STS	TDC DAC Trim Status

4.16.1 TDC_FB_DIV_INT_CNFG

TDC Feedback Divider Integer Configuration.

TDC_FB_DIV_INT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:0	tdc_fb_div_int	RW	0x24	TDC APLL Feedback Divider Integer. Integer portion of the TDC APLL feedback divider.

4.16.2 TDC_REF_DIV_CNFG

TDC Reference Divider Configuration.

TDC_REF_DIV_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:5	reserved	RW	0x0	Reserved
4	tdc_ref_sel	RW	0x0	TDC APLL Reference Select. Selects the reference for tdc apll 0x0 = osci/osco as tdc apll reference 0x1 = ref4 as tdc apll reference

TDC_REF_DIV_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
3	reserved	RO	0x0	Reserved
2:0	tdc_ref_div_config	RW	0x1	TDC Reference Divider Control. Controls the divide ratio of the TDC reference (either XO input or xcxo (single ended nCLKIN2), selected by tdc_ref_sel). This field should be programmed such that the reference to the TDC APLL is between 10MHz and 30MHz. 0x0 = Bypass divider. 0x1 = Divide by 2 0x2 = Divide by 4 0x3 = Divide by 8 0x4 = Divide by 16

4.16.3 TDC_CTRL

TDC Filter Status.

TDC_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1	tdc_dac_recal_req	RW	0x0	TDC DAC Recalibration Trigger. Triggers the TDC DAC recalibration when written to 1. This bit should be written to 0 before programming it to 1.
0	tdc_en	RW	0x0	TDC Enable. Controls whether the TDC is enabled. Must be enabled in Jitter Attenuator mode, when enabling a reference clock LOS or frequency monitor.

4.16.4 TDC_DAC_TRIM_CAL_STS

TDC DAC Trim Status.

TDC_DAC_TRIM_CAL_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1:0	tdc_dac_trim_sts	RO	0x0	TDC DAC Trim Status

4.17 DPLL

DPLL Registers.

Table 29. DPLL Register Index

Offset (Hex)	Register Module Base Address: 0x500	
	Register Name	**Register Description**
0x0	DPLL_REF_FB_CNFG	DPLL Reference Configuration
0x2	DPLL_REF_PRIORITY_CNFG	DPLL Mode Configuration
0x4	DPLL_MODE_CNFG	DPLL Mode Configuration
0x6	DPLL_XTAL_OFFSET_CNFG	DPLL XTAL Offset Configuration
0x7	DPLL_DECIMATOR_CNFG	DPLL Decimator Configuration
0x8	DPLL_BANDWIDTH_CNFG	DPLL Bandwidth Configuration
0xA	DPLL_DAMPING_CNFG	DPLL Damping Configuration
0xC	DPLL_PHASE_SLOPE_LIMIT_CNFG	DPLL Phase Slope Limit Configuration
0x10	DPLL_HOLDOVER_CNFG	DPLL Holdover Configuration
0x18	DPLL_INTEGRATOR_LIMIT_CNFG	DPLL Integrator Limit Configuration

Table 29. DPLL Register Index

Offset (Hex)	Register Module Base Address: 0x500	
	Register Name	Register Description
0x19	DPLL_FBDIV_CNFG	DPLL Feedback divider Configuration
0x1A	DPLL_HS_CNFG	DPLL Hitless Switching Configuration
0x1C	DPLL_WR_FREQ_PHASE_TIMER_CNFG	DPLL Write Frequency and Phase Timer enable
0x20	DPLL_PHASE_OFFSET_CNFG	DPLL Phase Offset Configuration
0x22	DPLL_TEMP_PHASE_OFFSET_CNFG	Temperature Sensor Enable
0x30	DPLL_FB_DIV_NUM_CNFG	DPLL Feedback Divider Numerator Configuration
0x38	DPLL_FB_DIV_DEN_CNFG	DPLL Feedback Divider Denominator Configuration
0x40	DPLL_FB_DIV_INT_CNFG	DPLL Feedback Divider Integer Configuration
0x48	DPLL_LOCK_CNFG	DPLL Lock Configuration
0x50	DPLL_CTRL	DPLL Control
0x51	DPLL_HOLDOVER_CTRL	DPLL Holdover Control
0x52	DPLL_HS_CTRL	DPLL Hitless Switching Control
0x53	DPLL_FILTER_DIS_CTRL	DPLL Filter Update Disable Control
0x58	DPLL_WR_PHASE_CTRL	DPLL Write Phase Control
0x60	DPLL_WR_FREQ_CTRL	DPLL Write Frequency Control
0x68	DPLL_TIMED_WR_FREQ_CTRL	DPLL Timed Write Frequency Control
0x70	DPLL_FB_WR_FREQ_CTRL	DPLL Timed Write Frequency Control
0x78	DPLL_EVENT	DPLL Events
0x80	DPLL_STS	DPLL Status
0x81	DPLL_LOL_CNT_STS	DPLL Loss of Lock Counter Status
0x82	DPLL_REF_STS	DPLL Reference Status
0x84	DPLL_WR_FREQ_PHASE_TIMER_STS	DPLL Timed Write Frequency and Phase Status
0x90	DPLL_FILTER_STS	DPLL Filter Status
0x88	DPLL_PHASE_STS	DPLL Phase Status

4.17.1 DPLL_REF_FB_CNFG

DPLL Reference Configuration.

DPLL_REF_FB_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:14	reserved	RO	0x0	Reserved
13:10	dpll_gpio_ref_sel_debounce_delay	RW	0x2	DPLL gpio reference select debounce delay. Determines the debounce delay in system clock cycles for dpll reference selection through gpio. A nonzero value results in the actual delay = 4 * value - 2 0x0 = Bypass
9:7	dpll_fb_sel	RW	0x4	DPLL feedback select. Selects the DPLL feedback source. 0x0 = REF 0 0x1 = REF 1 0x2 = REF 2 0x3 = REF 3 0x4 = clk_dpll_fbdiv 0x5 = time_clk_divided 0x6 = Reserved 0x7 = Reserved

DPLL_REF_FB_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
6	dpll_revertive_en	RW	0x0	DPLL revertive reference switch. No description 0x0 = Non-revertive 0x1 = Revertive
5	dpll_hitless_en	RW	0x0	DPLL hitless reference switch. No description 0x0 = Hitless disabled 0x1 = Hitless enabled
4:2	dpll_ref_sel	RW	0x0	DPLL manual reference clock selection. No description 0x0 = REF 0 0x1 = REF 1 0x2 = REF 2 0x3 = REF 3 0x4 = REF 4 0x5 = DPLL0 (clk_dpll1_fbdiv), DPLL1 (clk_dpll0_fbdiv), DPLL2 (clk_dpll0_fbdiv) 0x6 = DPLL0 (clk_dpll2_fbdiv), DPLL1 (clk_dpll2_fbdiv), DPLL2 (clk_dpll1_fbdiv) 0x7 = time_clk_divided
1:0	dpll_ref_sel_mode	RW	0x0	DPLL reference clock selection mode. No description 0x0 = Controlled by dpll_ref_sel 0x1 = GPI/GPIO 0x2 = Auto mode 0x3 = Reserved

4.17.2 DPLL_REF_PRIORITY_CNFG

DPLL Mode Configuration.

DPLL_REF_PRIORITY_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:12	reserved	RO	0x0	Reserved
11:10	dpll_ref3_priority	RW	0x0	Reference Clock 3 Priority. Sets the clock's priority for DPLL reference switching. If multiple references are set to the same priority level, they are prioritized from the lowest numbered (ref0) to the highest numbered (ref3). Priority of lower numbered references is only applied during initial reference selection. In revertive mode, the DPLL will not switch to a lower numbered reference with same priority as the currently qualified one. 0x0 = First priority 0x1 = Second priority 0x2 = Third priority 0x3 = Fourth priority
9:8	dpll_ref2_priority	RW	0x0	Reference Clock 2 Priority. Sets the clock's priority for DPLL reference switching. If multiple references are set to the same priority level, they are prioritized from the lowest numbered (ref0) to the highest numbered (ref3). Priority of lower numbered references is only applied during initial reference selection. In revertive mode, the DPLL will not switch to a lower numbered reference with same priority as the currently qualified one. 0x0 = First priority 0x1 = Second priority 0x2 = Third priority 0x3 = Fourth priority

DPLL_REF_PRIORITY_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:6	dpll_ref1_priority	RW	0x0	Reference Clock 1 Priority. Sets the clock's priority for DPLL reference switching. If multiple references are set to the same priority level, they are prioritized from the lowest numbered (ref0) to the highest numbered (ref3). Priority of lower numbered references is only applied during initial reference selection. In revertive mode, the DPLL will not switch to a lower numbered reference with same priority as the currently qualified one. 0x0 = First priority 0x1 = Second priority 0x2 = Third priority 0x3 = Fourth priority
5:4	dpll_ref0_priority	RW	0x0	Reference Clock 0 Priority. Sets the clock's priority for DPLL reference switching. If multiple references are set to the same priority level, they are prioritized from the lowest numbered (ref0) to the highest numbered (ref3). Priority of lower numbered references is only applied during initial reference selection. In revertive mode, the DPLL will not switch to a lower numbered reference with same priority as the currently qualified one. 0x0 = First priority 0x1 = Second priority 0x2 = Third priority 0x3 = Fourth priority
3	dpll_ref3_disable	RW	0x0	Reference Clock 3 Selection Disable. Controls whether reference clock 3 may be selected as the DPLL reference clock. 0x0 = Reference clock may be selected, subject to qualification by the Loss-of-Signal and Frequency monitors, and prioritization according to dpll_refx_priority 0x1 = Reference clock cannot be selected, ref_invalid_sts set to 1
2	dpll_ref2_disable	RW	0x0	Reference Clock 2 Selection Disable. Controls whether reference clock 2 may be selected as the DPLL reference clock. 0x0 = Reference clock may be selected, subject to qualification by the Loss-of-Signal and Frequency monitors, and prioritization according to dpll_refx_priority 0x1 = Reference clock cannot be selected, ref_invalid_sts set to 1
1	dpll_ref1_disable	RW	0x0	Reference Clock 1 Selection Disable. Controls whether reference clock 1 may be selected as the DPLL reference clock. 0x0 = Reference clock may be selected, subject to qualification by the Loss-of-Signal and Frequency monitors, and prioritization according to dpll_refx_priority 0x1 = Reference clock cannot be selected, ref_invalid_sts set to 1
0	dpll_ref0_disable	RW	0x0	Reference Clock 0 Selection Disable. Controls whether reference clock 0 may be selected as the DPLL reference clock. 0x0 = Reference clock may be selected, subject to qualification by the Loss-of-Signal and Frequency monitors, and prioritization according to dpll_refx_priority 0x1 = Reference clock cannot be selected, ref_invalid_sts set to 1

4.17.3 DPLL_MODE_CNFG

DPLL Mode Configuration.

DPLL_MODE_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15	relock_on_sync	RW	0x1	Force relock on resync. When set, forces the DPLL to lose lock upon a divider sync
14	man_bw_sel_ctrl	RW	0x0	Manual bandwidth select. When bw_sel_mode is 0x3, this field selects the bandwidth settings to use. 0x0 = Uses the settings from the acquire_bw_shift and acquire_bw_mult fields 0x1 = Uses the settings from the normal_bw_shift and normal_bw_mult fields

DPLL_MODE_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
13:12	bw_sel_mode	RW	0x0	Bandwidth mode select. This field selects the way in which the DPLL filter bandwidth is controlled. 0x0 = The filter bandwidth is controlled by the DPLL state machine 0x1 = The filter bandwidth is controlled by a GPIO (see gpio_func) 0x2 = Reserved 0x3 = The filter bandwidth is controlled by the man_bw_sel_ctrl field
11	reserved	RO	0x0	Reserved
10	gpio_mode_en	RW	0x0	DPLL mode control select. This bit is used to select whether the DPLL mode is controlled by the CSRs or by the GPIOs (see gpio_func) 0x0 = The DPLL mode is controlled by the dpll_mode CSR settings 0x1 = The DPLL mode is controlled by a GPIO (see gpio_func)
9:8	reserved	RO	0x0	Reserved
7	phase_source_sel	RW	0x0	DPLL filter phase source select. This bit selects the source of the filter phase data. 0x0=from decimator 0x1=from write_phase
6	bw_damp_sw	RW	0x1	Automatic bandwidth/damping switching. Enables the DPLL to switch to the Locking Loop Filter bandwidth and damping settings when the DPLL is in the Acquire state while locking. Refer to dpll_lock_timer. 0x0 = Always use Normal Operation settings. 0x1 = Use Locking settings when the DPLL is in the Acquire state.
5	auto_holdover_in_manual_en	RW	0x1	Auto holdover/freerun in manual mode. Controls whether the DPLL can go into Holdover/Freerun when in forced modes. 0x0 = Forced state does not change on LOS 0x1 = On LOS, forced acquire or forced locked changes to holdover/freerun depending on los_to_freerun
4	los_to_freerun	RW	0x0	Reference Loss-of-Signal to Freerun. Controls whether the DPLL enters Freerun or Holdover mode when the current reference clock is invalid. 0x0 = Holdover. 0x1 = Freerun.
3	reserved	RO	0x0	Reserved
2:0	dpll_mode	RW	0x6	DPLL mode selection. Selects DPLL mode. 0x0 = Forces the DPLL into the Freerun state. 0x1 = Forces the DPLL into the Normal state. 0x2 = Forces the DPLL into the Holdover state. 0x3 = Forces the DPLL into the Write Frequency state. 0x4 = Forces the DPLL into the Acquire state. 0x5 = Reserved 0x6 = Automatic mode. The usual setting for DPLL apps. 0x7 = Reserved

4.17.4 DPLL_XTAL_OFFSET_CNFG

DPLL XTAL Offset Configuration.

DPLL_XTAL_OFFSET_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:0	xtal_trim	RW	0x0	Crystal Trim Offset. Crystal fractional frequency offset compensation. This is an 8-bit 2's complement value. Resolution = 2-20 ~ 1 ppm, Range = +-2-13 ~ +-122 ppm. apll_fb_sdm_order must be set to a value greater than 0 for xtal_trim to operate correctly.

4.17.5 DPLL_DECIMATOR_CNFG

DPLL Decimator Configuration.

DPLL_DECIMATOR_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7	reserved	RO	0x0	Reserved
6:4	dec_hitless_bw_shift	RW	0x3	Hitless Switch Decimator Bandwidth. Shift to set the decimator bandwidth during a hitless reference switch or holdover-normal switch for measuring the phase offset. If dpll_hitless_en is set to zero, this field is ignored. After device startup, should only be changed while tdc_en is set to 0.
3:0	dec_bw_shift	RW	0x6	Decimator Bandwidth. Shift to set the decimator bandwidth. 0 puts the decimator in feedthrough (infinite bandwidth). After device startup, should only be changed while tdc_en is set to 0.

4.17.6 DPLL_BANDWIDTH_CNFG

DPLL Bandwidth Configuration.

DPLL_BANDWIDTH_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:11	acquire_bw_shift	RW	0x17	Acquire Loop Filter Bandwidth Shift. Coarse control of the DPLL loop filter bandwidth in the Acquire state while locking to the input clock. Default bandwidth = 1023 Hz.
10:8	acquire_bw_mult	RW	0x1	Acquire Loop Filter Bandwidth Multiplier. Fine control of the DPLL loop filter bandwidth in the Acquire state while locking to the input clock. Default bandwidth = 1023 Hz
7:3	normal_bw_shift	RW	0x10	Normal Operation Loop Filter Bandwidth Shift. Coarse control of the DPLL loop filter bandwidth in the Normal state when locked to the input clock. Default bandwidth = 127 Hz.
2:0	normal_bw_mult	RW	0x0	Normal Operation Loop Filter Bandwidth Multiplier. Fine control of the DPLL loop filter bandwidth in the Normal state when locked to the input clock. Default bandwidth = 127 Hz.

4.17.7 DPLL_DAMPING_CNFG

DPLL Damping Configuration.

DPLL_DAMPING_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:14	reserved	RO	0x0	Reserved
13:11	acquire_damping_shift	RW	0x5	Acquire Loop Filter Damping Shift. Coarse control of the DPLL loop filter damping in the Acquire state while locking to the input clock. Default damping causes 0.8dB peaking in the frequency domain jitter transfer function.
10:8	acquire_damping_mult	RW	0x1	Acquire Loop Filter Damping Multiplier. Fine control of the DPLL loop filter damping in the Acquire state while locking to the input clock. Default damping causes 0.8dB peaking in the frequency domain jitter transfer function.
7:6	reserved	RO	0x0	Reserved
5:3	normal_damping_shift	RW	0x2	Normal Operation Loop Filter Damping Shift. Coarse control of the DPLL loop filter damping in the Normal state when locked to the input clock. Default damping causes 0.2 dB peaking in the frequency domain jitter transfer function.
2:0	normal_damping_mult	RW	0x4	Normal Operation Loop Filter Damping Multiplier. Fine control of the DPLL loop filter damping in the Normal state when locked to the input clock. Default damping causes 0.2 dB peaking in the frequency domain jitter transfer function.

4.17.8 DPLL_PHASE_SLOPE_LIMIT_CNFG

DPLL Phase Slope Limit Configuration.

DPLL_PHASE_SLOPE_LIMIT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:27	reserved	RO	0x0	Reserved
26:0	phase_slope_limit	RW	0x7FFF FFF	Phase Slope Limit. Control of the phase slope limit of the output clocks. This represents the maximum instant relative frequency change of the output clock. This is an unsigned unitless number although it is often expressed as us/s or ns/s. It is recommended to program a value that is approx. 10% smaller than the required limit to leave some room for the integrator to adjust to frequency offsets. The resolution of 1 LSB is FC3: 2^-(32+6) = 3.64e-12 = 3.64 ps/s. FC3W: 2^-(27+6) = 1.16e-10 = 116 ps/s.

4.17.9 DPLL_HOLDOVER_CNFG

DPLL Holdover Configuration.

DPLL_HOLDOVER_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:31	holdover_sw_val	RW	0x0	Value to used to override the holdover filter value when manual_holdover is set to 1
30:18	reserved	RO	0x0	Reserved
17:15	holdover_bw_shift	RW	0x7	Holdover Filter Bandwidth Shift. Coarse control of the holdover bandwidth. A value of zero disables the holdover filter (infinite bandwidth).
14:12	holdover_bw_mult	RW	0x0	Holdover Filter Bandwidth Multiplier. Fine control of the holdover filter bandwidth. A value of zero disables the holdover filter (infinite bandwidth), which is also the default setting.
11:4	holdover_history	RW	0x0	Holdover history. Controls the age of the stored holdover value, in seconds. So a value of 1 means 1s, 2 means 2s, etc. A value of 0 results in using the instantaneous holdover value.
3:1	reserved	RO	0x0	Reserved
0	manual_holdover	RW	0x0	When set, enables holdover filter override.

4.17.10 DPLL_INTEGRATOR_LIMIT_CNFG

DPLL Integrator Limit Configuration.

DPLL_INTEGRATOR_LIMIT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:0	integrator_limit	RW	0xFF	Filter integral path limit.

4.17.11 DPLL_FBDIV_CNFG

DPLL Feedback divider Configuration.

DPLL_FBDIV_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:5	reserved	RO	0x0	Reserved
4	dpll_fbdiv_src_clk_sel	RW	0x0	DPLL0 feedback clock source selection. NOTE: This applies to DPLL0 only. When dpll_fb_sel is set to 0x4, this bit selects whether the clock source comes from the VCO divider or FOD0. For DPLL1 and DPLL2, when dpll_fb_sel is set to 0x4, the clock source only comes from FOD1 and FOD2. 0x0 = VCO div N 0x1 = FOD0
3:2	reserved	RO	0x0	Reserved
1	vco_dpll_fb_div_en	RW	0x0	DPLL0 only. VCO sourced DPLL Feedback Divider Clock Enable. DPLL0 only. Enables the VCO clock going to the DPLL feedback divider. May be set to 0 to reduce power consumption when an external feedback clock is selected with dpll_fb_sel. 0x0 = VCO Clock to DPLL0 feedback divider disabled, unused for DPLL1/2 0x1 = VCO Clock to DPLL0 feedback divider enabled, unused for DPLL 1/2
0	fod_dpll_fb_div_en	RW	0x0	FOD sourced DPLL Feedback Divider Clock Enable. Enables the FOD clock going to the DPLL feedback divider. May be set to 0 to reduce power consumption when an external feedback clock is selected with dpll_fb_sel. 0x0 = FOD Clock to DPLL feedback divider disabled 0x1 = FOD Clock to DPLL feedback divider enabled

4.17.12 DPLL_HS_CNFG

DPLL Hitless Switching Configuration.

DPLL_HS_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:10	reserved	RO	0x0	Reserved
9:8	hs_imm_clr_mode	RW	0x0	DPLL hitless clear mode. Bit 0: when set, disabling hitless switching immediately causes the built out phase offset to be cleared; when cleared, it takes effect on the next reference switch Bit 1: when set, clearing the latched hitless phase offset takes effect immediately; when cleared, it takes effect on the next reference switch
7:0	hs_counter_limit	RW	0x4	DPLL hitless counter limit. Sets the limit on the number of reference clock cycles before reading using the phase measurement of the decimator. This determines the wait time after the DPLL has switched to the hitless bandwidth config and would require a phase reading of the newly selected valid reference.

4.17.13 DPLL_WR_FREQ_PHASE_TIMER_CNFG

DPLL Write Frequency and Phase Timer enable.

DPLL_WR_FREQ_PHASE_TIMER_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:19	reserved	RO	0x0	Reserved
18	timer_write_frequency_ precise_sel	RW	0x0	When set, the timed_write_frequency value is applied exactly for the duration programmed in write_time_val, which means that it can take up to 1ms for it to take effect. When this bit is low, the timed_write_frequency value gets applied immediately when written.

DPLL_WR_FREQ_PHASE_TIMER_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
17	timer_en_write_frequency	RW	0x0	When set, the software-controlled frequency gets applied for the timer duration as defined in write_timer_val. When cleared, the software-controlled write frequency does not get used.
16	timer_en_write_phase	RW	0x0	When set, the software-controlled phase gets applied for the timer duration as defined in write_timer_val. When cleared, the software-controlled phase gets applied indefinitely.
15:0	write_timer_val	RW	0x0	Write frequency/timer duration. This value indicates the time during which the write_phase or timed_write_frequency value gets applied. The units are milliseconds.

4.17.14 DPLL_PHASE_OFFSET_CNFG

DPLL Phase Offset Configuration.

DPLL_PHASE_OFFSET_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:0	dpll_phase_offset	RW	0x0	DPLL Phase Offset. Manually adds phase offset between the reference and feedback clocks. This is a 16-bit 2's complement value. The resolution is the TDC resolution / 2 (~ 9.2 ps) and the range is approx. +-300ns. All outputs move together using this precision setting. This field is not used when hitless switching is enabled. This register is atomic. When the most significant byte (bits [15:8]) is written, the new value is applied to the DPLL.

4.17.15 DPLL_TEMP_PHASE_OFFSET_CNFG

Temperature Sensor Enable.

DPLL_TEMP_PHASE_OFFSET_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:4	reserved	RO	0x0	Reserved
3:0	temp_mult	RW	0x0	Temperature to phase multiplier. Signed multiplier value to the temperature value for the temperature phase offset. Effective DPLL phase_offset = (temp_sensor_value * temp_mult + dpll_phase_offset) * 9.2ps

4.17.16 DPLL_FB_DIV_NUM_CNFG

DPLL Feedback Divider Numerator Configuration.

DPLL_FB_DIV_NUM_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:48	reserved	RO	0x0	Reserved
47:0	fb_div_num	RW	0x0	Feedback Divider Numerator. DPLL feedback divide numerator value. Refer to fb_div_int for details. This register field is part of an atomic group consisting of fb_div_num, fb_div_den and fb_div_int. When the most significant byte (bits [47:40]) of fb_div_num or fb_div_den, or the most significant byte (bits 20:16]) of fb_div_int is written, the value of all these fields are applied to the DPLL.

4.17.17 DPLL_FB_DIV_DEN_CNFG

DPLL Feedback Divider Denominator Configuration.

DPLL_FB_DIV_DEN_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:48	reserved	RO	0x0	Reserved
47:0	fb_div_den	RW	0x800000	Feedback Divider Denominator. DPLL feedback divide denominator value. Refer to fb_div_int for details. Note: The MSB (bit 47) of fb_div_den must be a '1'. For an arbitrary fraction M/N, this may be accomplished by left shifting the denominator value N until the MSB becomes 1, and then left shifting the numerator value M by the same number of bits to obtain the fb_div_num value. This register field is part of an atomic group consisting of fb_div_num, fb_div_den and fb_div_int. When the most significant byte (bits [47:40]) of fb_div_num or fb_div_den, or the most significant byte (bits 20:16]) of fb_div_int is written, the value of all these fields are applied to the DPLL.

4.17.18 DPLL_FB_DIV_INT_CNFG

DPLL Feedback Divider Integer Configuration.

DPLL_FB_DIV_INT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:24	reserved	RO	0x0	Reserved
23:0	fb_div_int	RW	0xC8	DPLL Feedback Clock Divider Integer. NOTE: For DPLL1 and DPLL2, only the bottom 20 bits are used. DPLL feedback divide integer value. The DPLL feedback clock frequency must be no more than 33 MHz, and must be equal to the frequency of the reference clock divided by id_ratio, or equal to the reference clock when the input divider is bypassed by id_byp_en. This register field is part of an atomic group consisting of fb_div_num, fb_div_den and fb_div_int. When the most significant byte (bits [47:40]) of fb_div_num or fb_div_den, or the most significant byte (bits 20:16]) of fb_div_int is written, the value of all these fields are applied to the DPLL. Writing the MSB of fb_div_int causes the DPLL feedback divider and SDM to be reset.

4.17.19 DPLL_LOCK_CNFG

DPLL Lock Configuration.

DPLL_LOCK_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40	mask_dpll_lock_at_startup	RW	0x0	Mask dpll lock signal to reset controller. When set, and the DPLL is enabled, the reset controller will ignore the corresponding DPLL lock for purposes of enabling output clocks when out_startup is set to 3.
39:36	reserved	RO	0x0	Reserved
35:32	dpll_lol_cnt_thresh	RW	0x0	DPLL Loss-of-Lock Counter Threshold. While the DPLL Loss-of-Lock counter (dpll_lol_cnt) exceeds this threshold, the dpll_lol_lmt bit is set. The maximum valid value for this field is 14 (0xE).

DPLL_LOCK_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:16	dpll_lock_timer	RW	0xFF	DPLL lock timer. Specifies the time interval during which the absolute value of the phase detector error must remain below the DPLL lock threshold (dpll_lock_thresh) in order to declare lock. The DPLL switches from the Acquire state to the Normal state when the threshold has been met for half of this time interval. If enabled by bw_damp_sw, the loop filter bandwidth and damping settings revert at this time from the Acquire settings to the Normal settings. When the threshold has been met again for half of this time interval, the DPLL declares lock. The minimum value is 2. The units are ms.
15:0	dpll_lock_thresh	RW	0x155	DPLL lock threshold. Specifies the threshold that the absolute value of the phase detector error must remain below during the DPLL lock timer (dpll_lock_timer) in order to declare lock. The units are 8 * TDC resolution.

4.17.20 DPLL_CTRL

DPLL Control.

DPLL_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:5	reserved	RO	0x0	Reserved
4	filter_proportional_en	RW	0x1	Filter proportional enable
3	filter_integrator_en	RW	0x1	Filter integrator enable
2:1	filter_status_sel	RW	0x0	Filter status source selection. Chooses which values to load from the DPLL 0x0 = int_term 0x1 = prop_int_sum 0x2 = holdover_filter 0x3 = Reserved
0	dpll_en	RW	0x0	DPLL Enable. Controls whether the DPLL is enabled. 0x0 = DPLL is disabled. For synthesizer or DCO apps. 0x1 = DPLL is enabled. For jitter attenuator or synchronization apps.

4.17.21 DPLL_HOLDOVER_CTRL

DPLL Holdover Control.

DPLL_HOLDOVER_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1	holdover_filter_rst	RW	0x0	when set reset the holdover filter. This is a self clearing bit, and users are expected to only write 1 to the register.
0	holdover_history_rst	RW	0x0	when set, clears the holdover history. This is a self clearing bit, and users are expected to only write 1 to the register.

4.17.22 DPLL_HS_CTRL

DPLL Hitless Switching Control.

DPLL_HS_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	hs_offset_clr_b	RW	0x1	when low, clears the latched hitless phase offset, either immediately or on the next reference switch event (see hs_imm_clr_mode). The DPLL will not apply nor latch a new reference offset on a switch while this bit is low.

4.17.23 DPLL_FILTER_DIS_CTRL

DPLL Filter Update Disable Control.

DPLL_FILTER_DIS_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	filter_update_dis	RW	0x0	DPLL filter update disable. When set to 1, disables the filter_update calculations.

4.17.24 DPLL_WR_PHASE_CTRL

DPLL Write Phase Control.

DPLL_WR_PHASE_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:28	reserved	RO	0x0	Reserved
27:0	write_phase	RW	0x0	Software-controlled write phase value, signed. When this field is written, the phase value in this register gets applied at the input of the filter, assuming the phase_source_sel bit is high. If the timer_en_write_phase bit is set, it gets applied for write_timer_val ms. Otherwise, it gets applied indefinitely. Units are TDC resolution / 2

4.17.25 DPLL_WR_FREQ_CTRL

DPLL Write Frequency Control.

DPLL_WR_FREQ_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:33	reserved	RO	0x0	Reserved
32:0	write_freq	RW	0x0	Write Frequency. Frequency control word for synthesizer/DCO mode. This is a 33-bit 2's complement value. The units are 2^-44 * 1e6 [ppm]. The maximum setting is +-243ppm. apll_fb_sdm_order must be set to a value greater than 0 for write_freq to operate correctly. An update to an individual byte of this multi-byte register will take effect when the individual byte is written.

4.17.26 DPLL_TIMED_WR_FREQ_CTRL

DPLL Timed Write Frequency Control.

DPLL_TIMED_WR_FREQ_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:33	reserved	RO	0x0	Reserved
32:0	timed_write_frequency	RW	0x0	Timed software-controlled write frequency value. When this field is written, if the timer_en_write_frequency_cnfg bit is set, the frequency offset in this field gets applied to the DPLL for write_timer_val ms. When the timer expires, the frequency offset goes back to the one defined in DPLL_WRITE_FREQ_CTRL. The units are the same as the write_freq field

4.17.27 DPLL_FB_WR_FREQ_CTRL

DPLL Timed Write Frequency Control.

DPLL_FB_WR_FREQ_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:49	reserved	RO	0x0	Reserved
48	dpll_fb_scaler_en	RW	0x0	Feedback scaler enable. DPLL0 only. Enables the DPLL0 feedback scaler for the feedback SDM and divider. 0x0 = fb_scaler disabled/bypassed. 0x1 = fb_scaler enabled.
47:33	reserved	RO	0x0	Reserved
32:0	dpll_fb_write_freq	RW	0x0	Feedback write frequency. DPLL0 only. Frequency control word for the DPLL0 feedback SDM divider. This is a 33-bit 2's complement value. The units are 2^-44 * 1e6 [ppm]. The maximum setting is +-243ppm. The dpll_fb_scaler must be enabled for the write_freq to operate correctly. An update to an individual byte of this multi-byte register will take effect when the individual byte is written.

4.17.28 DPLL_EVENT

DPLL Events.

DPLL_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:5	reserved	RO	0x0	Reserved
4	dpll_bw_sel_ch_evt	RW1C	0x0	DPLL Bandwidth Select Change Event. Set to 1 when the DPLL bw_sel changes. This bit will always remain low if dpll_bw_sel_mode is 0x0 or 0x2.
3	dpll_state_ch_evt	RW1C	0x0	DPLL State Change event. Set to 1 when the DPLL state machine changes state. Once asserted, this bit will remain asserted until cleared by a write of '1' to this bit position.
2	dpll_holdover_evt	RW1C	0x0	DPLL Holdover event. Set to 1 when the DPLL state machine is in the holdover state. Once asserted, this bit will remain asserted until cleared by a write of '1' to this bit position, as long as the DPLL state machine is no longer in the holdover state.
1	dpll_lol_lmt_evt	RW1C	0x0	DPLL Loss-of-Lock Counter Threshold Exceeded status. Set while the DPLL Loss-of-Lock counter (dpll_lol_cnt) exceeds the threshold set in dpll_lol_cnt_thresh. This bit cannot be cleared by software while the condition persists (i.e., the dpll_lol_cnt should be cleared before this bit can be cleared). 0x0 = Loss-of-lock counter has not exceeded the threshold since the last time the bit was cleared 0x1 = Loss-of-lock counter exceeded the threshold since the last time the bit was cleared
0	dpll_lol_evt	RW1C	0x0	DPLL Loss-of-lock event. Set to 1 when the DPLL lock status transitions from locked to unlocked. Once asserted, this bit will remain asserted until cleared by a write of '1' to this bit position.

4.17.29 DPLL_STS

DPLL Status.

DPLL_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7	reserved	RO	0x0	Reserved
6:4	dpll_state_sts	RO	0x0	DPLL FSM state. Decode as follows: 0x0 = Freerun 0x1 = Normal / locked 0x2 = Holdover 0x3 = Write frequency 0x4 = Acquire 0x5 = Hitless switch
3	reserved	RO	0x0	Reserved
2:1	dpll_ref_sel_sts	RO	0x0	DPLL reference clock selection status. Indicates the reference clock selected by the DPLL. This 2-bit value refers to 8 reference options depending on the DPLL number as indicated below. For DPLL0: 0x0 = ref_div0 or ref4; 0x1 = ref_div1 or dpll1_fb; 0x2 = ref_div2 or dpll2_fb; 0x3 = ref_div3 or time_clock_div For DPLL1: 0x0 = ref_div0 or ref4; 0x1 = ref_div1 or dpll0_fb; 0x2 = ref_div2 or dpll2_fb; 0x3 = ref_div3 or time_clock_div For DPLL2: 0x0 = ref_div0 or ref4; 0x1 = ref_div1 or dpll0_fb; 0x2 = ref_div2 or dpll1_fb; 0x3 = ref_div3 or time_clock_div
0	dpll_lock_sts	RO	0x0	DPLL lock status. Indicates the DPLL lock status: 0x0 = Unlocked 0x1 = Locked

4.17.30 DPLL_LOL_CNT_STS

DPLL Loss of Lock Counter Status.

DPLL_LOL_CNT_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	reserved	RO	0x0	Reserved
3:0	dpll_lol_cnt	RW	0x0	DPLL Loss-of-Lock Counter. This counter increments each time the DPLL lock status deasserts, and saturates at 0xF. It is cleared by writing it to 0x0, and may be preset by writing the desired value.

4.17.31 DPLL_REF_STS

DPLL Reference Status.

DPLL_REF_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	reserved	RO	0x0	Reserved
3	ref3_invalid_sts	RO	0x0	DPLL reference 3 invalid status. Indicates whether reference clock 3 is currently considered to be invalid. This occurs if the clock is disqualified by one or more of the Loss-of-Signal and Frequency monitors, or ref_disable is set to 1. 0x0 = Clock is valid 0x1 = Clock is invalid

DPLL_REF_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
2	ref2_invalid_sts	RO	0x0	DPLL reference 2 invalid status. Indicates whether reference clock 2 is currently considered to be invalid. This occurs if the clock is disqualified by one or more of the Loss-of-Signal and Frequency monitors, or ref_disable is set to 1. 0x0 = Clock is valid 0x1 = Clock is invalid
1	ref1_invalid_sts	RO	0x0	DPLL reference 1 invalid status. Indicates whether reference clock 1 is currently considered to be invalid. This occurs if the clock is disqualified by one or more of the Loss-of-Signal and Frequency monitors, or ref_disable is set to 1. 0x0 = Clock is valid 0x1 = Clock is invalid
0	ref0_invalid_sts	RO	0x0	DPLL reference 0 invalid status. Indicates whether reference clock 0 is currently considered to be invalid. This occurs if the clock is disqualified by one or more of the Loss-of-Signal and Frequency monitors, or ref_disable is set to 1. 0x0 = Clock is valid 0x1 = Clock is invalid

4.17.32 DPLL_WR_FREQ_PHASE_TIMER_STS

DPLL Timed Write Frequency and Phase Status.

DPLL_WR_FREQ_PHASE_TIMER_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:0	timer_val_sts	RO	0x0	This field indicates the current value of the internal write phase/frequency timer. DPLL_WR_PHASE_CTRL or DPLL_TIMED_WR_FREQ_CTRL should only be written when this field reads as 0.

4.17.33 DPLL_FILTER_STS

DPLL Filter Status.

DPLL_FILTER_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:39	reserved	RO	0x0	Reserved
38:0	filter_sts	RO	0x0	DPLL Filter Status. Provides the integrator value from the filter.

4.17.34 DPLL_PHASE_STS

DPLL Phase Status.

DPLL_PHASE_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:28	reserved	RO	0x0	Reserved
27:0	phase_sts	RO	0x0	DPLL Phase status. Provides the phase data from the decimator. This is a 32-bit 2's complement value. The units are the TDC APLL VCO period divided by 62*8 (2.333ps for the nominal 864MHz).

4.18 XTALMON

XTAL Monitor Registers (xtalmon[0] = XIN, xtalmon[1]=REF4).

Table 30. XTALMON Register Index

Offset (Hex)	Register Module Base Address: 0x800	
	Register Name	Register Description
0x0	XTALMON_NOMINAL_MARGIN_CNFG	XTAL Monitor Nominal and Margin Configuration
0x4	XTALMON_WINDOW_CNFG	XTAL Monitor Miscellaneous Configuration
0x6	XTALMON_QUAL_CNFG	XTAL Monitor Qualification Configuration
0x7	XTALMON_CTRL	XTAL Monitor enable
0x8	XTALMON_EVENT	XTAL Monitor Events
0x9	XTALMON_CNT_EVENT	XTAL Counter Status
0xA	XTALMON_STS	XTAL Monitor Status

4.18.1 XTALMON_NOMINAL_MARGIN_CNFG

XTAL Monitor Nominal and Margin Configuration.

XTALMON_NOMINAL_MARGIN_CNFG Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
31:24	reserved	RO	0x0	Reserved
23:16	xtal_los_nom_num	RW	0x0	LOS Monitor Nominal Cycle Count. Sets the expected number of measuring clock periods within one monitor window. Measuring clock is the 216MHz divided TDC clock. A value of 0x0 is reserved and disables the LOS monitor. Disabling the monitor will cause the los_sts bit to be asserted.
15	reserved	RO	0x0	Reserved
14:8	xtal_los_acc_margin	RW	0x0	LOS Monitor Accept Threshold. An accepted clock monitoring window occurs when the final monitor counter value is within xtal_los_nom_num +/- xtal_los_acc_margin.
7	reserved	RO	0x0	Reserved
6:0	xtal_los_rej_margin	RW	0x0	LOS Monitor Reject Threshold. A rejected clock monitoring window occurs when the final monitor counter value is outside of xtal_los_nom_num +/- xtal_los_rej_margin.

4.18.2 XTALMON_WINDOW_CNFG

XTAL Monitor Miscellaneous Configuration.

XTALMON_WINDOW_CNFG Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
15:12	reserved	RO	0x0	Reserved
11:8	xtal_los_cnt_thresh	RW	0x0	Loss-of-Signal Counter Threshold. While the Loss-of-Signal counter (los_cnt) exceeds this threshold, the xtal_los_lmt_evt bit is set. The maximum valid value for this field is 30 (0x1E).
7	reserved	RO	0x0	Reserved

XTALMON_WINDOW_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
6:5	xtal_los_samp_clk_sel	RW	0x0	LOS sampling clock select. Selects the monitor's sampling clock 0x0 = Divided TDC clock 0x1 = FOD0 clock 0x2 = FOD1 clock 0x3 = FOD2 clock
4:0	xtal_los_div_ratio	RW	0x0	LOS Monitor Divide Ratio. This divide ratio must be set such that the monitored clock nominal frequency divided by xtal_los_div_ratio is less than 1/8 of the measuring clock frequency to achieve 25% accuracy. One period of the divided clock is the monitoring window duration. A value of 0 or 1 means divide by 1.

4.18.3 XTALMON_QUAL_CNFG

XTAL Monitor Qualification Configuration.

XTALMON_QUAL_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	xtal_los_good_times	RW	0x1	LOS Monitor Qualification Count. If this number of consecutive accepted clock LOS monitoring windows occur without a rejected window, then the clock is qualified and xtal_los_sts is set to 0. A value of 0 is reserved.
3:0	xtal_los_fail_times	RW	0x1	LOS Monitor Disqualification Count. If this number of rejected clock LOS monitoring windows occur without qualifying the clock, then the clock is disqualified and xtal_los_sts is set to 1. A value of 0 is reserved.

4.18.4 XTALMON_CTRL

XTAL Monitor enable.

XTALMON_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	xtal_los_mon_enable	RW	0x0	XTAL LOS Monitor Enable. Enables the XTAL LOS monitor. Monitor should be disabled while programming. 0x0 = XTAL LOS monitor disabled 0x1 = XTAL LOS monitor enabled

4.18.5 XTALMON_EVENT

XTAL Monitor Events.

XTALMON_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved

XTALMON_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
1	xtal_los_lmt_evt	RW1C	0x0	Loss-of-Signal Counter Threshold Exceeded status. Set while the Loss-of-Signal counter (xtal_los_cnt) exceeds the threshold set in los_cnt_thresh. This bit cannot be cleared by software while the condition persists (i.e., the xtal_los_cnt should be cleared before this bit can be cleared). 0x0 = Loss-of-signal counter has not exceeded the threshold since the last time the bit was cleared 0x1 = Loss-of-signal counter exceeded the threshold since the last time the bit was cleared
0	xtal_los_evt	RW1C	0x0	Loss-of-Signal Event status. Set while the clock monitor asserts LOS. This bit cannot be cleared by software while the LOS condition persists. This bit is set when the block comes out of reset and needs to be cleared after proper programming. 0x0 = Loss-of-signal not detected since the last time the bit was cleared 0x1 = Loss-of-signal detected since the last time the bit was cleared

4.18.6 XTALMON_CNT_EVENT

XTAL Counter Status.

XTALMON_CNT_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	reserved	RO	0x0	Reserved
3:0	xtal_los_cnt	RW	0x0	Loss-of-Signal Failure Counter. This counter increments each time the clock monitor asserts LOS, and saturates at 0xF. It is cleared by writing it to 0x0, and may be preset by writing the desired value. Preset may be used either as a debug tool or to cause a threshold alarm to happen sooner.

4.18.7 XTALMON_STS

XTAL Monitor Status.

XTALMON_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	xtal_los_sts	RO	0x1	Loss-of-Signal status. Current value of the LOS status from the clock monitor: 0x0 = Clock meets the monitoring criteria 0x1 = Loss-of-signal detected

4.19 LOSMON

LOS Monitor Registers.

Table 31. LOSMON Register Index

Offset (Hex)	Register Module Base Address: 0x810	
	Register Name	**Register Description**
0x0	LOSMON_NOMINAL_MARGIN_CNFG	LOS Monitor Nominal and Margin Configuration
0x8	LOSMON_WINDOW_CNFG	LOS Monitor Miscellaneous Configuration
0xA	LOSMON_QUAL_CNFG	LOS Monitor Qualification Configuration
0xB	LOSMON_CTRL	LOS Monitor enable
0xC	LOSMON_EVENT	LOS Monitor Events
0xD	LOSMON_CNT_EVENT	LOS Counter Status
0xE	LOSMON_STS	LOS Monitor Status

4.19.1 LOSMON_NOMINAL_MARGIN_CNFG

LOS Monitor Nominal and Margin Configuration.

LOSMON_NOMINAL_MARGIN_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:46	los_nom_num	RW	0x0	LOS Monitor Nominal Cycle Count. Sets the expected number of measuring clock periods within one monitor window. Measuring clock is the 216MHz divided TDC clock. A value of 0x0 is reserved and disables the LOS monitor. Disabling the monitor will cause the los_sts bit to be asserted, therefore the los_fail_mask bit should also be set when this field is written to 0x0.
45:34	reserved	RO	0x0	Reserved
33:17	los_acc_margin	RW	0x0	LOS Monitor Accept Threshold. An accepted clock monitoring window occurs when the final monitor counter value is within los_nom_num +/- los_acc_margin.
16:0	los_rej_margin	RW	0x0	LOS Monitor Reject Threshold. A rejected clock monitoring window occurs when the final monitor counter value is outside of los_nom_num +/- los_rej_margin.

4.19.2 LOSMON_WINDOW_CNFG

LOS Monitor Miscellaneous Configuration.

LOSMON_WINDOW_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:13	los_gap	RW	0x0	Gapped clock configuration. This field can be programmed so that the los monitor tolerates gapped clocks. The monitor will assert loss-of-signal when the final monitor counter value is outside of ((1+los_gap)*los_nom_num) +/- los_acc_margin. NOTE: This field in unused for losmon 4.
12:9	los_cnt_thresh	RW	0x0	Loss-of-Signal Counter Threshold. While the Loss-of-Signal counter (los_cnt) exceeds this threshold, the los_lmt_evt bit is set. The maximum valid value for this field is 30 (0x1E).
8:7	los_samp_clk_sel	RW	0x0	LOS sampling clock select. Selects the monitor's sampling clock 0x0 = Divided TDC clock 0x1 = FOD0 clock 0x2 = FOD1 clock 0x3 = FOD2 clock
6	reserved	RO	0x0	Reserved
5	los_fail_mask	RW	0x0	LOS Monitor Failure Mask. Masks the LOS monitor status contribution to ref_invalid_sts. NOTE: This field is unused for losmon 4. 0x0 = los_sts contributes to ref_invalid_sts 0x1 = los_sts does not contribute to ref_invalid_sts
4:0	los_div_ratio	RW	0x0	LOS Monitor Divide Ratio. This divide ratio must be set such that the monitored clock nominal frequency divided by los_div_ratio is less than 1/8 of the measuring clock frequency to achieve 25% accuracy. One period of the divided clock is the monitoring window duration. A value of 0 or 1 means divide by 1.

4.19.3 LOSMON_QUAL_CNFG

LOS Monitor Qualification Configuration.

LOSMON_QUAL_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	los_good_times	RW	0x1	LOS Monitor Qualification Count. If this number of consecutive accepted clock LOS monitoring windows occur without a rejected window, then the clock is qualified and los_sts is set to 0. A value of 0 is reserved.
3:0	los_fail_times	RW	0x1	LOS Monitor Disqualification Count. If this number of rejected clock LOS monitoring windows occur without qualifying the clock, then the clock is disqualified and los_sts is set to 1. A value of 0 is reserved.

4.19.4 LOSMON_CTRL

LOS Monitor enable.

LOSMON_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	los_mon_enable	RW	0x0	LOS monitor enable. Enables the LOS monitor. Monitor should be disabled while programming. Disabling the monitor causes los_sts to get asserted. 0 = LOS monitor disabled 1 = LOS monitor enabled

4.19.5 LOSMON_EVENT

LOS Monitor Events.

LOSMON_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1	los_lmt_evt	RW1C	0x0	Loss-of-Signal Counter Threshold Exceeded status. Set while the Loss-of-Signal counter (los_cnt) exceeds the threshold set in los_cnt_thresh. This bit cannot be cleared by software while the condition persists (i.e., the los_cnt should be cleared before this bit can be cleared). 0x0 = Loss-of-signal counter has not exceeded the threshold since the last time the bit was cleared 0x1 = Loss-of-signal counter exceeded the threshold since the last time the bit was cleared
0	los_evt	RW1C	0x0	Loss-of-Signal Event status. Set while the clock monitor asserts LOS. This bit cannot be cleared by software while the LOS condition persists. This bit is set when the block comes out of reset and needs to be cleared after proper programming. 0x0 = Loss-of-signal not detected since the last time the bit was cleared 0x1 = Loss-of-signal detected since the last time the bit was cleared

4.19.6 LOSMON_CNT_EVENT

LOS Counter Status.

LOSMON_CNT_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	reserved	RO	0x0	Reserved
3:0	los_cnt	RW	0x0	Loss-of-Signal Failure Counter. This counter increments each time the clock monitor asserts LOS, and saturates at 0xF. It is cleared by writing it to 0x0, and may be preset by writing the desired value. Preset may be used either as a debug tool or to cause a threshold alarm to happen sooner.

4.19.7 LOSMON_STS

LOS Monitor Status.

LOSMON_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1	ref_qual_sts	RO	0x0	Reference clock qualification status. Indicates whether the reference is considered failed due to disqualification by either the LOS or frequency monitors. 0x0 = Clock is disqualified 0x1 = Clock is qualified
0	los_sts	RO	0x1	Loss-of-Signal status. Current value of the LOS status from the clock monitor: 0x0 = Clock meets the monitoring criteria 0x1 = Loss-of-signal detected

4.20 FREQMON

Frequency Monitor Registers.

Table 32. FREQMON Register Index

Offset (Hex)	Register Module Base Address: 0x860	
	Register Name	**Register Description**
0x8	FREQMON_WINDOW_CNFG	FREQ Monitor Miscellaneous Configuration
0xC	FREQMON_NOMINAL_CNFG	FREQ Monitor Nominal Configuration
0x0	FREQMON_MARGIN_CNFG	FREQ Monitor Margin Configuration
0x10	FREQMON_CTRL	FREQ Monitor Enable
0x11	FREQMON_EVENT	FREQ Monitor Events
0x14	FREQMON_STS	FREQ Offset Status

4.20.1 FREQMON_WINDOW_CNFG

FREQ Monitor Miscellaneous Configuration.

FREQMON_WINDOW_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31	freq_fail_mask	RW	0x0	Frequency Monitor Failure Mask. Masks the frequency monitor status contribution to ref_invalid_sts. 0 = freq_fail_sts contributes to ref_invalid_sts 1 = freq_fail_sts does not contribute to ref_invalid_sts
30:29	freq_samp_clk_sel	RW	0x0	Frequency monitor sampling clock select. Selects the monitor's sampling clock 0x0 = Divided TDC clock 0x1 = FOD0 clock 0x2 = FOD1 clock 0x3 = FOD2 clock
28:0	freq_div_ratio	RW	0x2DC6C0	Frequency Monitor Divide Ratio. This divide ratio must be set to achieve the desired frequency monitoring time window. A longer monitoring window increases the tolerance of the monitor to frequency wander on references derived from a wide area network. A longer monitoring window also improves the frequency resolution of the monitor.A value of 0 or 1 means divide by 1 and is not valid.

4.20.2 FREQMON_NOMINAL_CNFG

FREQ Monitor Nominal Configuration.

FREQMON_NOMINAL_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:29	reserved	RO	0x0	Reserved
28:0	freq_nom_num	RW	0x0	Frequency Monitor Nominal Cycle Count. Sets the expected number of clock periods of the TDC ring oscillator frequency divided by 4 (nominally 216MHz) within one monitor window. A value of 0x0 is reserved and disables the frequency monitor. Disabling the monitor will cause the freq_fail_sts bit to be asserted, therefore the freq_fail_mask bit should also be set when this field is written to 0x0.

4.20.3 FREQMON_MARGIN_CNFG

FREQ Monitor Margin Configuration.

FREQMON_MARGIN_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:50	reserved	RO	0x0	Reserved
49:32	freq_acc_margin	RW	0x0	Frequency Monitor Accept Threshold. An accepted clock monitoring window occurs when the final monitor counter value is within freq_nom_num +/- freq_acc_margin. One accepted window qualifies the clock and freq_fail_sts is set to 0.
31:18	reserved	RO	0x0	Reserved
17:0	freq_rej_margin	RW	0x0	Frequency Monitor Reject Threshold. A rejected clock monitoring window occurs when the final monitor counter value is outside of freq_nom_num +/- freq_rej_margin. One rejected window disqualifies the clock and freq_fail_sts is set to 1.

4.20.4 FREQMON_CTRL

FREQ Monitor Enable.

FREQMON_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	freq_mon_enable	RW	0x0	Frequency monitor enable. Enables the freq monitor. Monitor should be disabled while programming. When disabled, this bit will reset the internal counters

4.20.5 FREQMON_EVENT

FREQ Monitor Events.

FREQMON_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1	freq_update_evt	RW1C	0x0	Frequency Monitor update event. This bit is set when a new frequency offset is available. It can be cleared by writing a 1, so the user can wait for a new reading to become available.
0	freq_fail_evt	RW1C	0x0	Frequency Monitor event status. Set while the frequency monitor disqualifies the clock. This bit cannot be cleared by software while the disqualified condition persists. This bit is set when the block comes out of reset and needs to be cleared after proper programming. 0x0 = Frequency monitor has not disqualified the clock since the last time the bit was cleared 0x1 = Frequency monitor has disqualified the clock since the last time the bit was cleared

4.20.6 FREQMON_STS

FREQ Offset Status.

FREQMON_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31	freq_fail_sts	RO	0x1	Frequency Monitor Status. Current value of the qualification status from the frequency monitor: 0x0 = Clock meets the monitoring criteria, clock qualified 0x1 = Failure detected, clock disqualified
30	reserved	RO	0x0	Reserved
29:0	freq_offset_sts	RO	0x3FFF FFFF	Frequency Offset Status. Signed cycle count offset from the nominal number measured by the frequency monitor, updated at the end of each monitoring window. It may be converted to ppm as follows: ppm offset = 1e6 * freq_offset_sts/(freq_nom_num - freq_offset_sts)

4.21 EEPROM

EEPROM Registers.

Table 33. EEPROM Register Index

Offset (Hex)	Register Module Base Address: 0x8E0	
	Register Name	**Register Description**
0x0	EEPROM_CNFG	EEPROM Configuration
0x8	EEPROM_ADDR_CNFG	EEPROM Address Configuration

Table 33. EEPROM Register Index

Offset (Hex)	Register Module Base Address: 0x8E0	
	Register Name	Register Description
0x9	EEPROM_EVENT	EEPROM Events
0xA	EEPROM_ERR_CNT_EVENT	EEPROM Error Counter Status

4.21.1 EEPROM_CNFG

EEPROM Configuration.

EEPROM_CNFG Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
63:56	reserved	RO	0x0	Reserved
55:44	eeprom_fall	RW	0x96	EEPROM falling edge time. Cycle number (counting down from eeprom_cycle) at which the SCL falling edge occurs. The default value is for a 60MHz clock and must be scaled according to the actual system clock frequency.
43:32	eeprom_rise	RW	0x1C2	EEPROM rising edge time. Cycle number (counting down from eeprom_cycle) at which the SCL rising edge occurs. The default value is for a 60MHz clock and must be scaled according to the actual system clock frequency.
31:20	eeprom_cycle	RW	0x258	EEPROM cycle time. Number of system clock cycles in one SCL period when running at 100kHz. The default value is for a 60MHz clock and must be scaled according to the actual system clock frequency.
19	reserved	RO	0x0	Reserved
18:16	eeprom_pad_drv_override	RW	0x3	SCL/GPIO0 and SDA/GPIO1 drv value overrides. SCL/GPIO0 and SDA/GPIO1 drv value overrides during EEPROM load. Similar encoding as the gpio_drv field
15:14	reserved	RO	0x0	Reserved
13	eeprom_ext_addr	RW	0x0	EEPROM extended address enable. Allows extended 10-bit addressing with a 1-byte I2C address, if supported by the EEPROM. 0x0 = The address is sent outside of the device address. 0x1 = Address bits 10:8 are sent in bits 2:0 of the device address, and address bits 7:0 are sent in the 1-byte address. eeprom_addr_size must be set to 0.
12	eeprom_addr_size	RW	0x1	EEPROM address size. Number of address bytes sent to the EEPROM during a read. 0x0 = 1-byte address 0x1 = 2-byte address
11:8	eeprom_length	RW	0x4	EEPROM size. Selects the number of bytes in the EEPROM for storing configurations. 0x0 = 128B 0x1 = 256B 0x2 = 512B 0x3 = 1KB 0x4 = 2KB 0x5 = 4KB 0x6 = 8KB 0x7 = 16KB 0x8 = 32KB 0x9 = 64KB
7	eeprom_gpio_load_en	RW	0x0	EEPROM gpio load enable. Enables loading of the common and/or user configurations from an external EEPROM device over GPIO0, used as SCL, and GPIO1, used as SDA. 0x0 = Disabled 0x1 = Enabled

EEPROM_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
6	eeprom_load_en	RW	0x0	EEPROM load enable. Enables loading of the common and/or user configurations from an external EEPROM device. The device loads configurations in the following order: OTP common, EEPROM common (if eeprom_load_en is set to 1), OTP user, and EEPROM user (if eeprom_load_en is set to 1). This bit may be programmed in the OTP common and/or user configurations to control whether the device attempts to load the common and/or user configurations from EEPROM. WARNING: If the SDA pin is held low or floating when the I2C master attempts to read the EEPROM, the I2C master will wait indefinitely until SDA becomes high before beginning the read request. 0x0 = Disabled 0x1 = Enabled
5:4	eeprom_i2c_speed	RW	0x0	EEPROM I2C speed. Selects the I2C master speed for EEPROM load. When the speed is 400kHz or 1MHz, eeprom_fall, eeprom_rise and eeprom_cycle are internally divided by 4 or 10 respectively to achieve the faster timing. The pad drive strength (eeprom_i2c_drv) should also be set according to the speed. 0x0 = 100kHz 0x1 = 400kHz 0x2 = 1MHz 0x3 = Reserved
3:0	eeprom_retry_count	RW	0x4	EEPROM Load Retry Count. Number of times to attempt to load EEPROM. If eeprom_bad or load failed (CRC error) after this number of attempts, the eeprom_load_fail status bit is set.

4.21.2 EEPROM_ADDR_CNFG

EEPROM Address Configuration.

EEPROM_ADDR_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7	reserved	RO	0x0	Reserved
6:0	eeprom_addr	RW	0x50	EEPROM Device Address. Sets the I2C device address of the EEPROM to load. (R.3.1.13.20)

4.21.3 EEPROM_EVENT

EEPROM Events.

EEPROM_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	reserved	RO	0x0	Reserved
3	eeprom_bad_evt	RW1C	0x0	EEPROM Not Detected. When high, indicates the EEPROM did not acknowledge a read access during device startup. In this case, eeprom_load_fail is not set. If EEPROM load is disabled (eeprom_load_en is set to 0), then the chip will not attempt to read the EERPOM and this bit cannot be set. Cleared by writing it to 1.
2	eeprom_config_empty_evt	RW1C	0x0	EEPROM Load of Empty Configuration. When high, indicates the EEPROM load attempted to load a configuration that did not select any blocks, during device startup. Cleared by writing it to 1.

EEPROM_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
1	eeprom_load_fail_evt	RW1C	0x0	EEPROM Load Failure. When high, indicates the EEPROM load failed during device startup. This bit is not set if the EEPROM does not respond (eeprom_bad is set instead). Cleared by writing it to 1.
0	eeprom_crc_err_evt	RW1C	0x0	EEPROM Load CRC Error. When high, indicates the EEPROM load encountered one or more CRC errors during device startup. Cleared by writing it to 1.

4.21.4 EEPROM_ERR_CNT_EVENT

EEPROM Error Counter Status.

EEPROM_ERR_CNT_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	reserved	RO	0x0	Reserved
3:0	eeprom_crc_err_cnt	RW	0x0	EEPROM CRC Error Counter. This counter increments each time the loader detects a CRC error while reading the EEPROM, and saturates at 0xF. It is cleared by writing it to 0x0, and may be preset by writing the desired value. Preset may be used as a debug tool.

4.22 OTP

OTP Registers.

Table 34. OTP Register Index

Offset (Hex)	Register Module Base Address: 0x900	
	Register Name	**Register Description**
0x30	OTP_EVENT	OTP Events

4.22.1 OTP_EVENT

OTP Events.

OTP_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RW1C	0x0	Reserved
1	otp_load_fail_evt	RW1C	0x0	OTP Load Failure. When high, indicates the OTP load failed during device startup. Cleared by writing it to 1.
0	otp_crc_err_evt	RW1C	0x0	OTP Load CRC Error. When high, indicates the OTP load encountered one or more CRC errors during device startup. Cleared by writing it to 1.

4.23 LDO

LDO Registers.

Table 35. LDO Register Index

Offset (Hex)	Register Module Base Address: 0x950	
	Register Name	**Register Description**
0x1D	FOD_LDO_CNFG	FOD0 LDO Configuration

4.23.1 FOD_LDO_CNFG

FOD0 LDO Configuration.

FOD_LDO_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:5	reserved	RO	0x0	Reserved
4:3	fod_ldo_config	RW	0x2	FOD1 LDO voltage select. FOD0 LDO voltage select. fod1_ldo_config sets LDO voltage. 0x0 = 1.236 0x1 = 1.266 0x2 = 1.291 0x3 = 1.313
2	reserved	RO	0x0	Reserved
1	fod_ldo_fast_strt	RW	0x0	FOD1 LDO fast startup select. FOD0 LDO fast startup select. 0x0 = Disable fast startup. Use during normal operation. 0x1 = Enable fast startup. Use during device power-up.
0	fod_en_ldo	RW	0x1	FOD LDO enable. To conserve power, the LDO for an unused FOD may be disabled. 0x0 = Disable 0x1 = Enable

4.24 TIME_SYNC_TOD

Timesync TOD and Clock Registers.

Table 36. TIME_SYNC_TOD Register Index

Offset (Hex)	Register Module Base Address: 0xA00	
	Register Name	**Register Description**
0x0	TIME_CLOCK_GEN_CNFG	Time Clock Count Configuration Register
0x2	TIME_SYNC_CNFG	Time Sync Configuration Register
0x4	SUB_SYNC_GEN_CNFG	Time Sync Count Configuration Register
0x8	TOD_ASYNC_ADJ_MAX_CNFG	TOD Max Adjust Configuration Register
0xA	TOD_ENABLE_CTRL	TOD Enable Register
0xB	TOD_COUNTER_UPDATE_CTRL	TOD Comparison Enable Register
0xC	TOD_ASYNC_ADJUST_VAL_CTRL	TOD Asynchronous Adjust Value Register
0xE	TOD_ASYNC_ADJUST_REQ_CTRL	TOD Asynchronous Adjust Request Register
0x10	TOD_SYNC_LOAD_VAL_CTRL	TOD Synchronous Load Counter Value Register
0x20	TOD_SYNC_LOAD_EN_CTRL	TOD Synchronous Load Enable Register
0x21	TOD_SYNC_LOAD_REQ_CTRL	TOD Synchronous Load Request Register
0x30	TOD_COMPARE_VAL_CTRL	TOD Comparison Value Register
0x40	TOD_COMPARE_EN_CTRL	TOD Comparison Enable Register
0x48	TOD_COMPARE_EVENT	TOD Compare Status Register
0x5F	TOD_COUNTER_READ_REQ	TOD Counter Read Request
0x60	TOD_COUNTER_STS	TOD Counter Status Register

4.24.1 TIME_CLOCK_GEN_CNFG

Time Clock Count Configuration Register.

TIME_CLOCK_GEN_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:10	reserved	RO	0x0	Reserved
9:8	time_clock_src	RW	0x2	Time Clock Source Config. Indicates the time clock source. 0x0 = FOD0 0x1 = FOD1 0x2 = FOD2 0x3 = Reserved
7:6	reserved	RO	0x0	Reserved
5:0	time_clock_gen_count	RW	0x13	Time Clock Generation Counter Config. Indicates the number of DCO clock cycles (minus 1) in a Time Clock cycle. A value of 0 indicates bypass (Time Clock is the same as the DCO clock). Defaults to 19 (25MHz for a 500MHz DCO clock)

4.24.2 TIME_SYNC_CNFG

Time Sync Configuration Register.

TIME_SYNC_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:12	reserved	RO	0x0	Reserved
11:8	time_sync_del	RW	0x0	Time Sync Delay Config. Indicates the delay of the time_ sync signals with regards to the time_clock signal, in DCO clock cycles. Signed value. Defaults to 0. This value is limited to -5 to +5. In addition, it cannot be set to a value greater than time_clock_gen_count.
7:6	reserved	RO	0x0	Reserved
5:0	time_sync_width	RW	0x10	Time Sync Pulse Width Config. Indicates the width of the Time Sync. Note that the pulse width needs to be less than the time sync period minus one DCO clock or one Time Clock (depending on whether the width is specified in DCO cycles or Time Clock cycles). No protection will be provided against values that do not respect this rule. 0x0 = 5 DCO clock cycles 0x1 = 50 DCO clock cycles 0x2 = 500 DCO clock cycles 0x3 = 5K DCO clock cycles 0x4 = 50K DCO clock cycles 0x5 = 500K DCO clock cycles 0x6 = 5M DCO clock cycles 0x10 = 1 time clock cycle 0x11 = 2 time clock cycle 0x12 = 4 time clock cycle

TIME_SYNC_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
time_sync_width (continued)				0x13 = 8 time clock cycle 0x14 = 16 time clock cycle 0x15 = 32 time clock cycle 0x16 = 64 time clock cycle 0x17 = 128 time clock cycle 0x18 = 256 time clock cycle 0x19 = 512 time clock cycle 0x1A = 1024 time clock cycle 0x1B = 2048 time clock cycle 0x1C = 4096 time clock cycle
time_sync_width (continued)				0x1D = 8192 time clock cycle 0x1E = 16,384 time clock cycle 0x1F = 32,768 time clock cycle

4.24.3 SUB_SYNC_GEN_CNFG

Time Sync Count Configuration Register.

SUB_SYNC_GEN_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31	reserved	RO	0x0	Reserved
30:0	sub_sync_count	RW	0xC34	Time Clock Generation Counter Config. Indicates the number of Time Clock cycles (minus 1) in a Time Sync cycle. Defaults to 3124 (8kHz for a 25MHz Time Clock)

4.24.4 TOD_ASYNC_ADJ_MAX_CNFG

TOD Max Adjust Configuration Register.

TOD_ASYNC_ADJ_MAX_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:12	reserved	RO	0x0	Reserved
11:0	async_adjust_max	RW	0xC8	Maximum Async Adjust Value. Maximum async adjust allowed. This value is unsigned and should represent the number or Time Clock cycles in 8us (8us being the maximum PRD adjustment requirement) Defaults to 200 (8us @ 25MHz)

4.24.5 TOD_ENABLE_CTRL

TOD Enable Register.

TOD_ENABLE_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	enable	RW	0x0	Time Sync TOD Block Enable. Used to enable this block. Unless enabled, the counters do not count. This bit should be set only once the configuration fields have all be programmed.

4.24.6 TOD_COUNTER_UPDATE_CTRL

TOD Comparison Enable Register.

TOD_COUNTER_UPDATE_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	tod_counter_update_dis	RW	0x0	TOD Counter Update Disable. When set, the fields in the TOD_COUNTER_STS register do not get updated every microsecond. This allows manual atomic read control.

4.24.7 TOD_ASYNC_ADJUST_VAL_CTRL

TOD Asynchronous Adjust Value Register.

TOD_ASYNC_ADJUST_VAL_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:13	reserved	RO	0x0	Reserved
12:0	async_adjust_val	RW	0x0	Asynchronous TOD Counter Adjust Value. Asynchronous adjust value for the Sub Sync Counter. This value is signed and represents a number of Time Clock cycles adjustment. Should never bet set to a value greater than async_adjust_max_cnfg of the Time Sync cycle.

4.24.8 TOD_ASYNC_ADJUST_REQ_CTRL

TOD Asynchronous Adjust Request Register.

TOD_ASYNC_ADJUST_REQ_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	async_adjust_req	RW1S	0x0	Asynchronous TOD Counter Adjust Request. On a transition from low to high, the async_adjust_val_ctrl will be added to the Sub Sync Counter, as soon as it is possible to do so. This bit goes back to zero when the adjust has been applied.

4.24.9 TOD_SYNC_LOAD_VAL_CTRL

TOD Synchronous Load Counter Value Register.

TOD_SYNC_LOAD_VAL_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
127:83	reserved	RO	0x0	Reserved
82:31	sync_counter_load_val	RW	0x0	Synchronous Sync Load Value. When doing a synchronous load, this value gets loaded to the Sync counter. Defaults to 0.
30:0	sub_sync_counter_load_val	RW	0x0	Synchronous Sub-Sync Load Value. When doing a synchronous load, this value gets loaded to the Sub Sync counter. Defaults to 0.

4.24.10 TOD_SYNC_LOAD_EN_CTRL

TOD Synchronous Load Enable Register.

TOD_SYNC_LOAD_EN_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1	sync_load_enable	RW	0x1	Synchronous TOD Sync Counter Load Enable. When set, this bit enables the load to the Sync Counter (when a sync_load_req is issued)
0	sub_sync_load_enable	RW	0x1	Synchronous TOD Sub-Sync Counter Load Enable. When set, this bit enables the load to the Sub-Sync Counter (when a sync_load_req is issued)

4.24.11 TOD_SYNC_LOAD_REQ_CTRL

TOD Synchronous Load Request Register.

TOD_SYNC_LOAD_REQ_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	sync_load_req	RW1S	0x0	Synchronous TOD Counter Load Request. On a transition from low to high, the block will wait for the next Time Sync edge and synchronously load the value from the sub_ sync_count_load_val_ctrl and sync_count_load_val_ctrl into their respective counters, if their corresponding sync_load_enable bits are set. This bit can only be set by software, and gets cleared when the load has been done.

4.24.12 TOD_COMPARE_VAL_CTRL

TOD Comparison Value Register.

TOD_COMPARE_VAL_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
127:83	reserved	RO	0x0	Reserved
82:31	tod_sync_compare_val	RW	0x0	TOD Comparison Value. The combined value of tod_sub_sync_compare_val and tod_sync_compare_val gets compared to the combined internal values of the Sub Sync and Sync counters and when they match, a pulse gets sent out indicating this match. This value is not synchronized to the Time Clock domain, so it needs to be programmed while tod_compare_enable is low. The comparison circuit will only start comparing once tod_compare_enable is high.
30:0	tod_sub_sync_compare_val	RW	0x0	TOD Sub-Sync Comparison Value. The combined value of tod_sub_sync_compare_val and tod_sync_compare_val gets compared to the combined internal values of the Sub Sync and Sync counters and when they match, a pulse gets sent out indicating this match. This value is not synchronized to the Time Clock domain, so it needs to be programmed while tod_compare_enable is low. The comparison circuit will only start comparing once tod_compare_enable is high. This should only be used when the Time Clock frequency is TBD MHz (100?) or less.

4.24.13 TOD_COMPARE_EN_CTRL

TOD Comparison Enable Register.

TOD_COMPARE_EN_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	tod_compare_enable	RW1S	0x0	TOD Comparison Enable. When high, this bit enables the comparison of the internal TOD counter against the value programmed in tod_compare_val_ctrl. This bit should be set only once tod_compare_val_ctrl has been programmed. Once the internal TOD has reached the tod_compare_val_ctrl value, this bit will go back to zero.

4.24.14 TOD_COMPARE_EVENT

TOD Compare Status Register.

TOD_COMPARE_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	tod_compare_evt	RW1C	0x0	TOD Comparison Event. This bit goes to one when the TOD counter reaches the TOD Compare Value. This bit remains high until written to one.

4.24.15 TOD_COUNTER_READ_REQ

TOD Counter Read Request.

TOD_COUNTER_READ_REQ Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	tod_counter_rd_req	RO	0x0	TOD Counter Read Req. A burst read to this location followed by the TOD_COUNTER_STS bytes ensures the updates are blocked and the data in TOD_COUNTER_STS can be read atomically. If this location is not read first, or if the tod_counter_update_dis bit is not set, the TOD_COUNTER_STS register could potentially get updated while it is getting read out (it is internally updated every microsecond).

4.24.16 TOD_COUNTER_STS

TOD Counter Status Register.

TOD_COUNTER_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
127:83	reserved	RO	0x0	Reserved
82:31	sync_counter_value	RO	0x0	Sync Portion of the TOD Counter. Binary value representing the time in Time Sync periods (as configured, e.g., 0.5Hz, 1Hz, 1kHz, etc.). The Sub Sync Counter gets updated every microsecond (as programmed in COUNT_1US). If the tod_counter_rd_req is read first, or if the tod_counter_update_dis bit is set, the Sub Sync Counter and Sync Counter value are guaranteed to be consistent with one another.
30:0	sub_sync_counter_value	RO	0x0	Sub Sync Portion of the TOD Counter. Binary value representing the time offset from the bit<31> in Time Clock periods. The Sub Sync Counter gets updated every microsecond (as programmed in COUNT_1US). If the tod_counter_rd_req is read first, or if the tod_counter_update_dis bit is set, the Sub Sync Counter and Sync Counter value are guaranteed to be consistent with one another.

4.25 TIME_SYNC_LPF

Timesync LPF Registers.

Table 37. TIME_SYNC_LPF Register Index

Offset (Hex)	Register Module Base Address: 0xA80	
	Register Name	Register Description
0x0	LPF_MODE_CNFG	Time Sync Channel Filter Configuration Register
0x1	LPF_BW_CNFG	Time Sync Channel Filter Bandwidth Configuration Register
0x2	LPF_DAMP_CNFG	Time Sync Channel Filter Damping Configuration Register
0x4	LPF_HOLDOVER_CNFG	Time Sync Channel Holdover Configuration Register
0x8	LPF_WR_FREQ_PHASE_TIMER_CNFG	Time Sync Channel Write Timer Configuration Register
0x10	LPF_LIMITS_CNFG	Time Sync Channel Filter Limits Configuration Register
0x18	LPF_CTRL	Time Sync Channel Control Register
0x19	LPF_DIS_CTRL	Time Sync Channel Filter Disable Register
0x1F	LPF_HOLDOVER_FILTER_RST_CTRL	Time Sync Channel Holdover Filter Reset Register
0x20	LPF_HOLDOVER_CTRL	Time Sync Channel Holdover Control Register
0x28	LPF_WR_PHASE_CTRL	Time Sync Channel Write Phase Register
0x30	LPF_WR_FREQ_CTRL	Time Sync Channel Write Frequency Register
0x38	LPF_TIMED_WR_FREQ_CTRL	Time Sync Channel Timed Write Frequency Register
0x40	LPF_FILTER_STS	Time Sync Channel Filter Status Register
0x48	LPF_WR_FREQ_PHASE_TIMER_STS	Time Sync Channel Write Timer Status Register

4.25.1 LPF_MODE_CNFG

Time Sync Channel Filter Configuration Register.

LPF_MODE_CNFG Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
7:3	reserved	RO	0x0	Reserved
2:0	lpf_mode	RW	0x0	This field controls the mode of the Time Sync Filter. 0x0 = Filter is disabled 0x1 = Places the filter into Write Phase mode. 0x2 = Forces the DPLL into Holdover mode 0x3 = Forces the filter into Write Frequency mode. 0x4 = Reserved 0x5 = Reserved 0x6 = Reserved 0x7 = Reserved

4.25.2 LPF_BW_CNFG

Time Sync Channel Filter Bandwidth Configuration Register.

LPF_BW_CNFG Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
7:3	lpf_bw_shift	RW	0xB	Coarse control of the filter bandwidth in the Normal state when locked to the input clock. Default bandwidth = 127 Hz.
2:0	lpf_bw_mult	RW	0x0	Fine control of the filter bandwidth in the Normal state when locked to the input clock. Default bandwidth = 127 Hz.

4.25.3 LPF_DAMP_CNFG

Time Sync Channel Filter Damping Configuration Register.

LPF_DAMP_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:6	reserved	RO	0x0	Reserved
5:3	lpf_damp_shift	RW	0x0	Coarse control of the filter damping in the Normal state when locked to the input clock. Default damping causes 0.1 dB peaking in the frequency domain jitter transfer function.
2:0	lpf_damp_mult	RW	0x0	Fine control of the filter damping in the Normal state when locked to the input clock. Default damping causes 0.1 dB peaking in the frequency domain jitter transfer function.

4.25.4 LPF_HOLDOVER_CNFG

Time Sync Channel Holdover Configuration Register.

LPF_HOLDOVER_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:14	reserved	RO	0x0	Reserved
13:11	lpf_hold_bw_shift	RW	0x7	Coarse control of the holdover bandwidth. A value of zero disables the holdover filter (infinite bandwidth).
10:8	lpf_hold_bw_mult	RW	0x0	Fine control of the holdover filter bandwidth. A value of zero disables the holdover filter (infinite bandwidth), which is also the default setting.
7:0	lpf_hold_history	RW	0x0	Controls the age of the stored holdover value, in seconds. So a value of 1 means 1s, 2 means 2s, etc. A value of 0 results in using the instantaneous holdover value.

4.25.5 LPF_WR_FREQ_PHASE_TIMER_CNFG

Time Sync Channel Write Timer Configuration Register.

LPF_WR_FREQ_PHASE_TIMER_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:19	reserved	RO	0x0	Reserved
18	timer_write_frequency_precise_sel	RW	0x0	When set, the timed_write_frequency value is applied exactly for the duration programmed in write_time_val, which means that it can take up to 1ms for it to take effect. When this bit is low, the timed_write_frequency value gets applied immediately when written.
17	lpf_timer_en_write_freq	RW	0x0	When set, the software-controlled frequency gets applied for the timer duration as defined in write_timer_val. When cleared, the software-controlled write frequency does not get used.
16	lpf_timer_en_write_phase	RW	0x0	When set, the software-controlled phase gets applied for the timer duration as defined in write_timer_val. When cleared, the software-controlled phase gets applied indefinitely.
15:0	lpf_write_timer_val	RW	0x0	This value indicates the time during which the write_phase or timed_write_frequency value gets applied. The units are milliseconds.

4.25.6 LPF_LIMITS_CNFG

Time Sync Channel Filter Limits Configuration Register.

LPF_LIMITS_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:59	reserved	RO	0x0	Reserved
58:32	lpf_phase_slope_lmt	RW	0x7FFF FFF	Control of the phase slope limit of the output clocks. This represents the maximum instant relative frequency change of the output clock. This is an unsigned unitless number although it is often expressed as us/s or ns/s. It is recommended to program a value that is approx. 10% smaller than the required limit to leave some room for the integrator to adjust to frequency offsets. The resolution of 1 LSB is 2-35 = 2.91e-11 = 29.1 ps/s.
31:8	reserved	RO	0x0	Reserved
7:0	lpf_integrator_lmt	RW	0xFF	filter integral path limit

4.25.7 LPF_CTRL

Time Sync Channel Control Register.

LPF_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:6	reserved	RO	0x0	Reserved
5	lpf_filter_integrator_en	RW	0x1	
4	lpf_filter_proportional_en	RW	0x1	
3:2	lpf_filter_status_sel	RW	0x0	This bit selects the source of the lpf_filter_sts value 0x0 = Integrator value 0x1 = Frequency offset 0x2 = Holdover frequency 0x3 = Reserved
1	reserved	RO	0x0	Reserved
0	lpf_en	RW	0x0	This bit enables the Time Sync Channel Loop Filter

4.25.8 LPF_DIS_CTRL

Time Sync Channel Filter Disable Register.

LPF_DIS_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	lpf_filter_dis	RW	0x1	When set, this bit disables the filter calculations. This bit has to be set to 1 whenever any filter configuration value gets updated, and cleared afterwards.

4.25.9 LPF_HOLDOVER_FILTER_RST_CTRL

Time Sync Channel Holdover Filter Reset Register.

LPF_HOLDOVER_FILTER_RST_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1	lpf_hold_filter_rst	WO	0x0	When written to 1, this bit resets the holdover filter.
0	lpf_hold_history_rst	WO	0x0	When written to 1, this bit resets the holdover history.

4.25.10 LPF_HOLDOVER_CTRL

Time Sync Channel Holdover Control Register.

LPF_HOLDOVER_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:37	reserved	RO	0x0	Reserved
36	lpf_manual_hold_en	RW	0x0	When set, manual holdover value is used for the holdover frequency.
35:33	reserved	RO	0x0	Reserved
32:0	lpf_hold_sw_val	RW	0x0	This value is used for the holdover frequency when lpf_manual_hold_en is set.

4.25.11 LPF_WR_PHASE_CTRL

Time Sync Channel Write Phase Register.

LPF_WR_PHASE_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:28	reserved	RO	0x0	Reserved
27:0	lpf_wr_phase	RW	0x0	Write Phase

4.25.12 LPF_WR_FREQ_CTRL

Time Sync Channel Write Frequency Register.

LPF_WR_FREQ_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:33	reserved	RO	0x0	Reserved
32:0	lpf_wr_freq	RW	0x0	Frequency control word for synthesizer/DCO mode. This is a 33-bit 2's complement value. The units are 2^-44 * 1e6 [ppm]. FOD has to be running in fractional mode to operate correctly

4.25.13 LPF_TIMED_WR_FREQ_CTRL

Time Sync Channel Timed Write Frequency Register.

LPF_TIMED_WR_FREQ_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:33	reserved	RO	0x0	Reserved
32:0	lpf_timed_wr_freq	RW	0x0	When this field is written, if the timer_en_write_frequency_cnfg bit is set, the frequency offset in this field gets applied to the DPLL for write_timer_val ms. When the timer expires, the frequency offset goes back to the one defined in LPF_WR_FREQ_CTRL. The units are the same as the write_freq field

4.25.14 LPF_FILTER_STS

Time Sync Channel Filter Status Register.

LPF_FILTER_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:43	reserved	RO	0x0	Reserved
42:40	filter_state_sts	RO	0x0	Time Sync LPF FSM state. Decode as follows: 0x0 = Freerun 0x1 = Normal 0x2 = Holdover 0x3 = Write frequency 0x4 = Acquire 0x5 = Hitless switch
39	reserved	RO	0x0	Reserved
38:0	lpf_filter_sts	RO	0x0	This field is used to read the filter value, as selected by lpf_filter_status_sel

4.25.15 LPF_WR_FREQ_PHASE_TIMER_STS

Time Sync Channel Write Timer Status Register.

LPF_WR_FREQ_PHASE_TIMER_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:0	lpf_timer_val_sts	RO	0x0	This field indicates the current value of the internal write phase/frequency timer. LPF_WR_PHASE_CTRL or LPF_WR_FREQ_CTRL should only be written when this field reads as 0.

4.26 TIME_SYNC_TDC

Timesync TDC Registers.

Table 38. TIME_SYNC_TDC Register Index

Offset (Hex)	Register Module Base Address: 0xB00	
	Register Name	**Register Description**
0x0	TIME_CLOCK_TDC_FANOUT_CNFG	Time Sync Channel Measurement Fanout Enable Configuration Register
0x4	TIME_CLOCK_MEAS_CNFG	Time Sync Channel Measurement Configuration Register
0x8	TIME_CLOCK_MEAS_DIV_CNFG	Time Sync Channel Measurement Divider Configuration Register
0xC	DPLL_FB_TDC_DIV_CNFG	DPLL FB to TDC Divider Configuration Register
0x10	TIME_CLOCK_MEAS_CTRL	Time Sync Channel Measurement Control Register
0x11	TDC_FIFO_COUNT_CTRL	Time Sync Channel Measurement FIFO Control Register
0x12	TDC_FIFO_CTRL	Time Sync Channel Measurement FIFO Control Register
0x2F	TDC_FIFO_READ_REQ	Time Sync Channel Measurement FIFO Read Request Register
0x30	TDC_FIFO_READ	Time Sync Channel Measurement FIFO Read Value Register
0x38	TDC_FIFO_STS	Time Sync Channel Measurement FIFO Status Register
0x39	TDC_FIFO_EVENT	Time Sync Channel Measurement FIFO Event Register

4.26.1 TIME_CLOCK_TDC_FANOUT_CNFG

Time Sync Channel Measurement Fanout Enable Configuration Register.

TIME_CLOCK_TDC_FANOUT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:29	reserved	RO	0x0	Reserved
28:27	time_sync_fine_dly	RW	0x0	Time clock fine delay. This setting allows for adding fine delay to the time_sync signal going to the TDC block. The delay added corresponds to 25ps x time_sync_fine_dly
26:25	time_sync_coarse_dly	RW	0x1	Time sync coarse delay. This setting is used to adjust the delay of the time_clock signal to the TDC to match the internal delays of other signals. This setting is used to adjust the delay of the time_sync signal to the TDC to match the internal delays of other signals. 0x0 = This setting adds a delay equivalent to a loopback from the internal time sync, going out the device, back in through an input, to the TDC block (this does not take into account external output load). 0x1 = This setting adds a delay equivalent to the delay from an input to the TDC block, or equivalent to the delay from a DPLL feedback divider to the TDC block. 0x2 = This setting is the same as 0x1, with an additional 75ps delay. 0x3 = This setting is the same as 0x1, with 120ps subtracted.
24	time_sync_dly_byp	RW	0x1	Time sync delay bypass. When set, no additional delay is added to the time_sync signal going to the internal TDC measurement block.
23:21	reserved	RO	0x0	Reserved
20:19	time_clock_fine_dly	RW	0x0	Time clock fine delay. This setting allows for adding fine delay to the time_clock signal going to the TDC block. The delay added corresponds to 25ps x time_clock_fine_dly
18:17	time_clock_coarse_dly	RW	0x1	Time clock coarse delay. This setting is used to adjust the delay of the time_clock signal to the TDC to match the internal delays of other signals., value decode 0x0 = This setting adds a delay equivalent to a loopback from the internal time clock, going out the device, back in through an input, to the TDC block (this does not take into account external output load). 0x1 = This setting adds a delay equivalent to the delay from an input to the TDC block, or equivalent to the delay from a DPLL feedback divider to the TDC block. 0x2 = This setting is the same as 0x0, with an additional 75ps delay. 0x3 = This setting is the same as 0x1, with 120ps subtracted
16	time_clock_dly_byp	RW	0x1	Time clock delay bypass. When set, no additional delay is added to the time_clock signal going to the internal TDC measurement block.
15:12	coarse_clk_mux_sel	RW	0x3	Coarse clock mux select. This field selects the source for the coarse clock source. The only valid option for this field is time_ref (0x3) 0x0 = Reserved 0x1 = Reserved 0x2 = Reserved 0x3 = time_ref 0x4 = Reserved 0x5 = Reserved 0x6 = Reserved 0x7 = Reserved

TIME_CLOCK_TDC_FANOUT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
11:8	sig2_mux_sel	RW	0x0	Second signal mux select. This field selects the source for the second signal used in the TDC measurement. 0x0 = REF<0> 0x1 = REF<1> 0x2 = REF<2> 0x3 = REF<3> 0x4 = ref_clk<5> 0x5 = ref_clk<6> 0x6 = dpll_fb_to_tdc (as selected by dpll_fb_tdc_mux_sel, undivided) 0x7 = dpll_fb_divided_to_tdc (as selected by dpll_fb_tdc_mux_sel, divided) 0x8 = time_clk_divided 0x9 = time_sync
7:4	sig1_mux_sel	RW	0x5	First signal mux select. This field selects the source for the first signal used in the TDC measurement. 0x0 = REF<0> 0x1 = REF<1> 0x2 = REF<2> 0x3 = REF<3> 0x4 = ref_clk<5> 0x5 = ref_clk<6> 0x6 = dpll_fb_to_tdc (as selected by dpll_fb_tdc_mux_sel, undivided) 0x7 = dpll_fb_divided_to_tdc (as selected by dpll_fb_tdc_mux_sel, divided) 0x8 = time_clk_divided 0x9 = time_sync
3	reserved	RO	0x0	Reserved
2	time_clk_divided_to_tdc_en	RW	0x0	Enables the fanout buffer for sending the time_clk_divided to the TDC
1	time_clk_to_tdc_en	RW	0x0	Enables the fanout buffer for sending the time_clk to the TDC
0	time_sync_to_tdc_en	RW	0x0	Enables the fanout buffer for sending the time_sync to the TDC

4.26.2 TIME_CLOCK_MEAS_CNFG

Time Sync Channel Measurement Configuration Register.

TIME_CLOCK_MEAS_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1	tdc_allow_neg	RW	0x0	Allow negative measurement. When set, this bit allows negative measurements (i.e., if the sig1 edge is slightly earlier than the sig2 edge, the measurement returned will be a small negative number). When cleared, the result would be positive and represent almost a full sig1 period. 0x0 = Disallow negative measurements 0x1 = Allow negative measurements
0	tdc_meas_mode	RW	0x0	Measurement Mode. Indicates whether we should be doing one-shot or continuous measurements. 0x0 = Continuous 0x1 = One-shot

4.26.3 TIME_CLOCK_MEAS_DIV_CNFG

Time Sync Channel Measurement Divider Configuration Register.

TIME_CLOCK_MEAS_DIV_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:30	reserved	RO	0x0	Reserved
29:24	time_ref_div	RW	0x13	Divide ratio for time ref. Indicates the divide ratio of the time_ref coarse clock for TDC measurements, minus 1. A value of 0 disables this clock and should be used only when using an external coarse clock. The Time Ref clock should run at a frequency between 10MHz and 33MHz (higher is better).
23:22	reserved	RO	0x0	Reserved
21:0	time_clock_meas_div	RW	0x0	Divide Ratio for TDC Measurements. Indicates the divide ratio of the time_clock for TDC measurements, minus 1. Defaults to 0 (no division). Can be used to slow down the TDC measurements when the two signals being compared are clocks. Unless actually measuring the internal time_clock against the external time clock, this field should be kept at 0. This field should be set to a value less than or equal to the sub_sync_count value. The sub_sync_count value also needs to be an integer multiple of this field.

4.26.4 DPLL_FB_TDC_DIV_CNFG

DPLL FB to TDC Divider Configuration Register.

DPLL_FB_TDC_DIV_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:24	reserved	RO	0x0	Reserved
23	dpll_fb_tdc_div_enb	RW	0x0	DPLL FB TDC Divider Enable. This bit is active low. 0x0 = Enable 0x1 = Disable
22	dpll_fb_tdc_div_setb	RW	0x0	DPLL FB TDC Divider Set. When this bit = 0, the divider is halted (held in set mode) and the divider output will not toggle. 0x0 = Set 0x1 = Not set
21	reserved	RO	0x0	Reserved
20	dpll_fb_tdc_div_byp_en	RW	0x1	DPLL FB TDC Divider Bypass. When this bit is = 1, the divider is bypassed and the divider input clock is passed directly to the Time Sync TDC for an effective divide ratio of 1. The frequency of the clock passed to the Time Sync TDC must not exceed 33MHz. 0x0 = Divider is not bypassed 0x1 = Divider is bypassed
19:0	dpll_fb_tdc_div_pgm	RW	0x0	DPLL FB TDC Divider ratio. The divider input clock frequency is divided by this value. The minimum divide value is 2. The divider may be bypassed for an effective divide ratio of 1. The frequency of the clock passed to the Time Sync TDC must not exceed 33MHz.

4.26.5 TIME_CLOCK_MEAS_CTRL

Time Sync Channel Measurement Control Register.

TIME_CLOCK_MEAS_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1	tdc_meas_start	RW1S	0x0	Measurement Request. Writing a 1 to this register starts the TDC measurements. If the block is configured to operate in continuous mode, this bit will remain high unless the measurement mode gets changed to one-shot. If the block is operating in one-shot mode, this bit will go back to zero when the measurement has been done.
0	tdc_meas_en	RW	0x0	Measurement Enable. Time Sync TDC Measurement Enable. Set to 1 to enable this block. 0x0 = Disabled 0x1 = Enabled

4.26.6 TDC_FIFO_COUNT_CTRL

Time Sync Channel Measurement FIFO Control Register.

TDC_FIFO_COUNT_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:5	reserved	RO	0x0	Reserved
4:0	fifo_count_compare	RW	0x0	Every time a count_alarm number of FIFO values has been stored, the count_alarm_evt bit gets set. A value of 0 disables this comparison.

4.26.7 TDC_FIFO_CTRL

Time Sync Channel Measurement FIFO Control Register.

TDC_FIFO_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	fifo_clear	RW	0x0	TDC FIFO clear request. When written to 1, all FIFO entries get cleared. This bit self clears.

4.26.8 TDC_FIFO_READ_REQ

Time Sync Channel Measurement FIFO Read Request Register.

TDC_FIFO_READ_REQ Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	fifo_read_req	RO	0x0	No physical bit, just used to send a trigger to read the FIFO

4.26.9 TDC_FIFO_READ

Time Sync Channel Measurement FIFO Read Value Register.

TDC_FIFO_READ Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:40	reserved	RO	0x0	Reserved
39:0	fifo_read_val	RO	0x0	TDC FIFO read value. Bits[39:13] represent the coarse count (i.e., the number of time clock pulses between the two sync signals). Bits[12:0] represent the fine count (i.e., the offset in TDC resolution). Both values are signed. Nominally, the offset is calculated with the following equation, assuming a 25MHz time clock and a 864MHz TDC: offset = (tdc_coarse_meas * 40e-9) + (tdc_fine_meas / (864e6*62));

4.26.10 TDC_FIFO_STS

Time Sync Channel Measurement FIFO Status Register.

TDC_FIFO_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1	fifo_full	RO	0x0	When set, indicates the FIFO is full
0	fifo_empty	RO	0x1	When set, indicates the FIFO is empty.

4.26.11 TDC_FIFO_EVENT

Time Sync Channel Measurement FIFO Event Register.

TDC_FIFO_EVENT Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1	fifo_overrun_evt	RW1C	0x0	When set, indicates the FIFO has been overrun (i.e., values have been written to it when it was full). Note that when this happens, the oldest values get overwritten.
0	fifo_count_alarm_evt	RW1C	0x0	When set, indicates that count_compare values have been written to the FIFO.

4.27 TIME_SYNC_TDC_FIFO_DBG

Timesync TDC FIFO Debug Registers.

Table 39. TIME_SYNC_TDC_FIFO_DBG Register Index

Offset (Hex)	Register Module Base Address: 0xC00	
	Register Name	**Register Description**
0x0	TDC_FIFO_PTR_STS	Time Sync Channel Pointer Debug Register
0x8	TDC_FIFO_ENTRY_0	
0x10	TDC_FIFO_ENTRY_1	
0x18	TDC_FIFO_ENTRY_2	
0x20	TDC_FIFO_ENTRY_3	
0x28	TDC_FIFO_ENTRY_4	
0x30	TDC_FIFO_ENTRY_5	
0x38	TDC_FIFO_ENTRY_6	
0x40	TDC_FIFO_ENTRY_7	

Table 39. TIME_SYNC_TDC_FIFO_DBG Register Index

Offset (Hex)	Register Module Base Address: 0xC00	
	Register Name	Register Description
0x48	TDC_FIFO_ENTRY_8	
0x50	TDC_FIFO_ENTRY_9	
0x58	TDC_FIFO_ENTRY_10	
0x60	TDC_FIFO_ENTRY_11	
0x68	TDC_FIFO_ENTRY_12	
0x70	TDC_FIFO_ENTRY_13	
0x80	TDC_FIFO_ENTRY_14	
0x88	TDC_FIFO_ENTRY_15	

4.27.1 TDC_FIFO_PTR_STS

Time Sync Channel Pointer Debug Register.

TDC_FIFO_PTR_STS Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
7:4	write_pointer	RO	0x0	Current value of the FIFO write pointer
3:0	read_pointer	RO	0x0	Current value of the FIFO read pointer

4.27.2 TDC_FIFO_ENTRY_0

TDC_FIFO_ENTRY_0 Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_0	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.3 TDC_FIFO_ENTRY_1

TDC_FIFO_ENTRY_1 Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_1	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.4 TDC_FIFO_ENTRY_2

TDC_FIFO_ENTRY_2 Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_2	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.5 TDC_FIFO_ENTRY_3

TDC_FIFO_ENTRY_3 Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_3	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.6 TDC_FIFO_ENTRY_4

TDC_FIFO_ENTRY_4 Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_4	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.7 TDC_FIFO_ENTRY_5

TDC_FIFO_ENTRY_5 Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_5	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.8 TDC_FIFO_ENTRY_6

TDC_FIFO_ENTRY_6 Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_6	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.9 TDC_FIFO_ENTRY_7

TDC_FIFO_ENTRY_7 Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_7	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.10 TDC_FIFO_ENTRY_8

TDC_FIFO_ENTRY_8 Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_8	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.11 TDC_FIFO_ENTRY_9

TDC_FIFO_ENTRY_9 Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_9	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.12 TDC_FIFO_ENTRY_10

TDC_FIFO_ENTRY_10 Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_10	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.13 TDC_FIFO_ENTRY_11

TDC_FIFO_ENTRY_11 Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_11	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.14 TDC_FIFO_ENTRY_12

TDC_FIFO_ENTRY_12 Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_12	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.15 TDC_FIFO_ENTRY_13

TDC_FIFO_ENTRY_13 Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_13	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.16 TDC_FIFO_ENTRY_14

TDC_FIFO_ENTRY_14 Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_14	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.27.17 TDC_FIFO_ENTRY_15

TDC_FIFO_ENTRY_15 Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:41	reserved	RO	0x0	Reserved
40:0	tdc_fifo_entry_15	RO	0x0	Nth value in the TDC FIFO. Bit[40] is the empty/full indication

4.28 REINIT

THIS SECTION CONSISTS OF ALIASES TO THE GLOBAL CONTROL REGISTERS.

Table 40. REINIT Register Index

Offset (Hex)	Register Module Base Address: 0xD00	
	Register Name	**Register Description**
0x0	SOFT_RESET_CTRL	Soft Reset Control
0x4	DIVIDER_SYNC_CTRL	Divider sync control

4.28.1 SOFT_RESET_CTRL

Soft Reset Control.

SOFT_RESET_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1	apll_reinit	RW	0x0	APLL Reinitialization. Writing this bit to 1 re-starts the startup sequence from the VCO calibration step, including divider synchronization.
0	soft_reset	RW	0x0	Soft Reset. Write '1' to trigger a soft reset which re-starts the reset sequence from the VCO calibration step. This bit will self-clear. Depending on the value of latch_inputs, inputs will be relatched to enable changing of the I2C address only (RevA feature).

4.28.2 DIVIDER_SYNC_CTRL

Divider sync control.

DIVIDER_SYNC_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7	vco_iod_sync	RW1S	0x0	Enables the manual resynch of VCO based IODs and DPLL0 FBDIV when it's source is the VCO. Enables the manual resync of VCO based IODs and DPLL0 FBDIV when it's source is the VCO This bit is high until the resync is done.
6:4	fod_iod_sync	RW1S	0x0	Enables the manual resync of FODs and FOD sourced IODs as well as DPLL FBDIVS when source is the FOD. Enables the manual resync of FODs and FOD sourced IODs as well as DPLL FBDIVS when source is the FOD Each bit will remain high until the corresponding FOD has been resynced.
3:1	reserved	RO	0x0	Reserved
0	divider_sync	RW	0x0	Divider Synchronization. Write '1' to trigger synchronization of DPLL and output dividers. The VCO-based IODs and DPLL0 FBDIV (if driven from VCO) are done first, followed by FOD0 and DPLL0 FBDIV (if driven from FOD0), followed by FOD1 and finally by FOD2. This bit auto-clears.

5. Revision History

Revision	Date	Description
1.03	Aug 20, 2025	▪ Updated the descriptions of the following register fields: write_freq and dpll_fb_write_freq ▪ Added the following register fields: tdc_dac_recal_req, fod_ldo_config, and fod_ldo_fast_strt
1.02	Jul 8, 2025	▪ Removed otp_manual_rdy_int_en register bit from INT_EN_CTRL ▪ Removed otp_manual_rdy_int_sts register bit from INT_STS ▪ Updated descriptions in INT_EN_CTRL and INT_STS ▪ Added APLL_EVENT and APLL_LOL_EVENT ▪ Updated description of dpll_fb_sel ▪ Updated description of dpll_ref_sel_sts ▪ Added OTP
1.01	Jun 4, 2025	▪ Updated Figure 1, Figure 2, Figure 3, and Figure 4 ▪ Updated Figure 6 and Figure 7 ▪ Changed Device Frequency Reference to APLL Frequency Reference throughout ▪ Updated Analog PLL section ▪ Updated Table 3 ▪ Updated DPLL0, DPLL1 and DPLL2, and Time Sync DCO ▪ Added Clock Output Enable section ▪ Updated Figure 15 and Figure 16 ▪ Added SYSREFsection ▪ Removed out_init_bias_cal ▪ Updated iod_phase_config ▪ Updated SYSREF_CNFG, SYSREF_CTRL ▪ Updated pad_gpio_oe_b ▪ Updated INPUT_DIV_CNFG ▪ Added TDCAPLL ▪ Updated fod_en_ldo ▪ Completed other minor changes throughout
1.00	Jul 8, 2024	Initial release

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD-PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers who are designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only to develop an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third-party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising from your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Disclaimer Rev.1.01)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit www.renesas.com/contact-us/.

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.