RC32312, RC32308

FemtoClock®3 Jitter Attenuator and Multi-Frequency Clock Synthesizer

The RC32312/RC32308 is an ultra-low phase noise jitter attenuator, multi-frequency synthesizer, synchronous Ethernet synchronizer, and digitally controlled oscillator (DCO). This flexible, low-power device outputs clocks with 25fs RMS jitter supporting 112Gbps and 224Gbps SerDes.

This document provides programming information for software engineers to use the RC32312/RC32308.

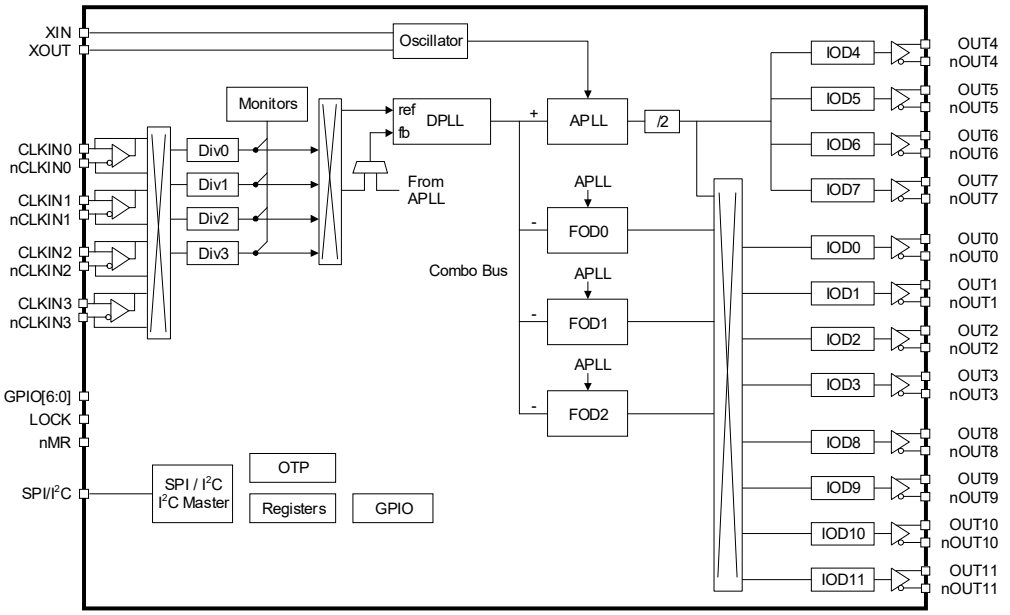


Figure 1. RC32312 Block Diagram

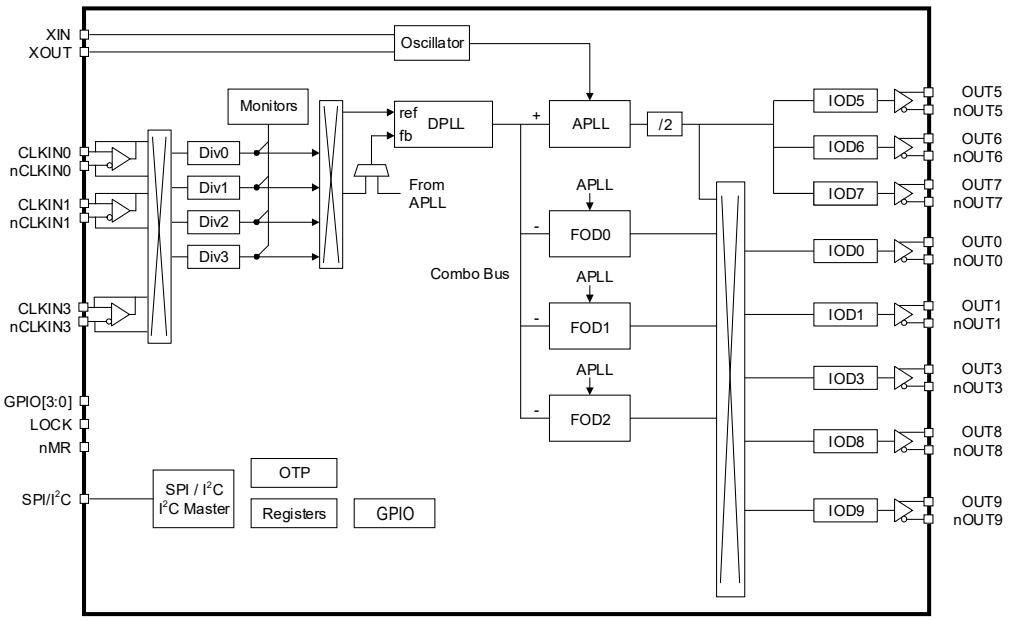


Figure 2. RC32308 Block Diagram

Note: Device specifications and ordering information are available in the *RC32312/RC32308 Datasheet*.

Contents

Figures

Tables

1. Functional Description

1.1 Overview

The RC32312/RC32308 is an ultra-low phase noise jitter attenuator, multi-frequency synthesizer, synchronous Ethernet synchronizer, and digitally controlled oscillator (DCO). This flexible, low-power device outputs clocks with 25fs RMS jitter supporting 112Gbps and 224Gbps SerDes.

The RC32312 has four differential clock inputs (CLKINx) and 12 differential clock outputs (OUTx). The RC32308 has three differential clock inputs (CLKINx) and eight differential clock outputs. Both devices provide a digital PLL (DPLL), an ultra-low phase noise synthesizer based on an analog PLL (APLL), and three low-phase noise synthesizers based on fractional output dividers (FOD). See Figure 1 for a block diagram of the RC32312, and see Figure 2 for a block diagram of the RC32308.

Each CLKINx can be configured as two single-ended inputs (CLKINx and nCLKINx). The clock inputs can operate at frequencies up to 1GHz when differential, and up to 250MHz when single-ended. Each OUTx can be configured as LVDS or HCSL outputs that can operate at frequencies up to 1GHz. Each OUTx can be configured as two LVCMOS outputs (OUTx and nOUTx) that can operate at frequencies up to 250MHz. When configured for LVCMOS, the OUTx and nOUTx pins can operate in-phase or 180° out-of-phase.

Figure 3 shows a detailed functional block diagram of RC32312. This diagram is also representative of the RC32308's architecture.

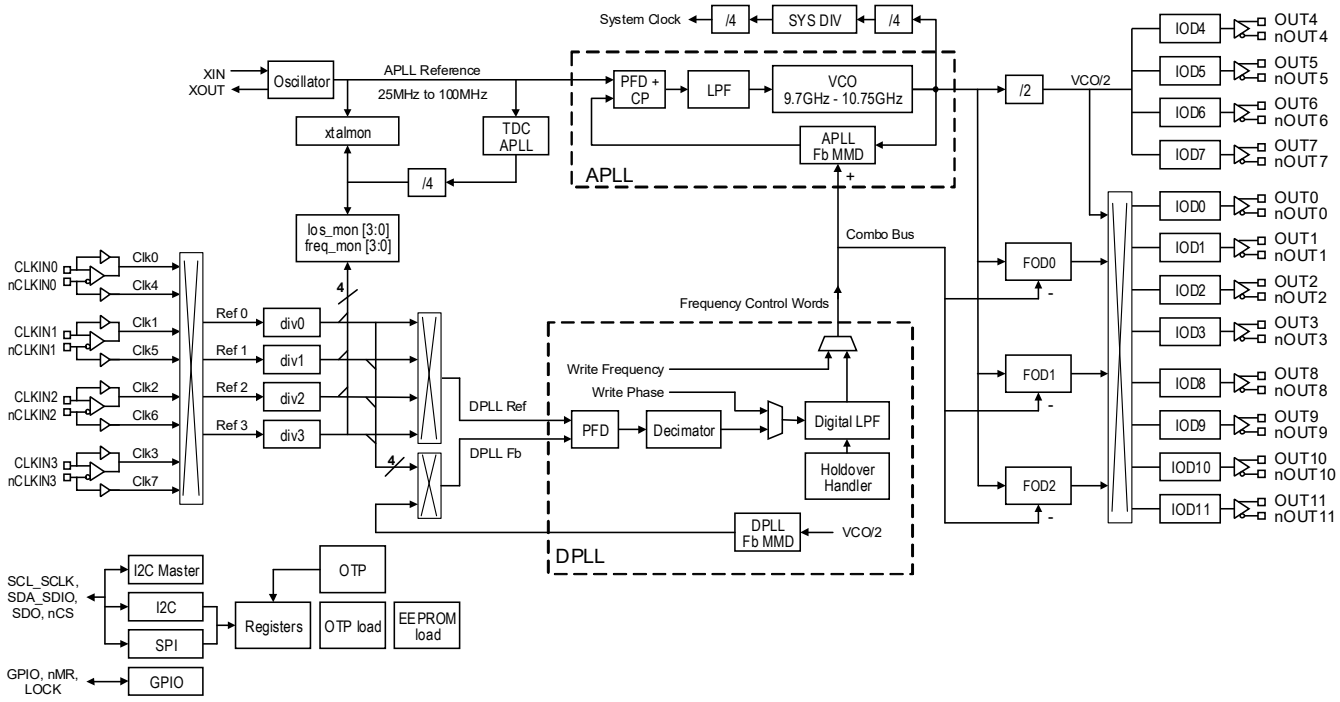


Figure 3. RC32312 Functional Block Diagram

1.2 Device Frequency Reference

The RC32312/RC32308 requires a frequency reference. The frequency reference can be implemented with an external crystal resonator and the device 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 *RC32312/RC32308 Datasheet*.

The device frequency reference must support the phase noise, frequency accuracy, and frequency stability requirements of the intended application. The phase noise of the frequency reference affects the phase noise of clocks output by the device.

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

For DCO applications, the accuracy of the frequency reference determines the frequency accuracy of the freerun 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 the system can support.

1.3 Analog PLL

The APLL is configured using registers in the APLL section.

The internal APLL locks to the device frequency reference and generates an ultra-low phase noise clock of virtually any frequency between 9.70GHz and 10.75GHz. The voltage controlled oscillator (VCO) is the synthesizer for the APLL and it outputs a frequency equal to the APLL reference frequency multiplied by the APLL feedback divide ratio. The APLL feedback divide ratio is programmed as indicated in the following equation using the following register fields: apll_fb_div_int and apll_fb_div_frac.

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

The APLL is the source of the DPLL feedback clock (VCO/2), and the DPLL steers the FFO of the APLL using frequency control words (FCW) as shown in the following equation (see also Figure 3). When the DPLL is in the Write Frequency State, the FCW is controlled by the write_freq register field. When the DPLL is in the Write Phase Mode, the FCW is the result of the values written to the write_phase register field after filtering by the DPLL low-pass filter.

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

The VCO clock is pre-divided by 2 and is available as an input to the integer output dividers (IOD). The pre-divided VCO clock is available directly to IOD[7:4], it is also available, via cross connect, to IOD[3:0] and IOD[11:8]. The undivided VCO clock is supplied to FOD[3:0].

1.4 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 integer and fractional division. The FOD output clocks are available, via cross connect, to IOD[3:0] and IOD[11:8].

The FODs have the following three modes of operation: synthesizer mode, DCO mode, and synchronous mode. All three FOD modes divide the VCO clock, the three modes differ in their intended applications, how the divide ratios are specified and how the FOD follows frequency steering applied to the VCO. The FOD mode is selected using the fod_sync_mode and fod_integer_mode register bits as indicated in Table 1.

Table 1. FOD Modes and Division Types

FOD Mode	Division Type	fod_sync_mode	fod_integer_mode
Synthesizer / DCO	Fractional	0x0	0x0
Synchronous	Fractional	0x1	0x0
	Integer	X	0x1

1.4.1 FOD Synthesizer Mode

In synthesizer mode, 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 will counteract digital frequency steering applied to the VCO by the DPLL, or by external software via the write_phase or write_freq register field. In this way an FOD based synthesizer is virtually unaffected by digital frequency steering applied to the VCO.

1.4.2 FOD Digitally Controlled Oscillator Mode

In DCO mode, 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.

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

An FOD in DCO mode will counteract digital frequency steering applied to the VCO by the DPLL, or by external software via the write_phase or write_freq register field. In this way an FOD based DCO is virtually unaffected by digital frequency steering applied to the VCO.

1.4.3 FOD Synchronous Mode

In synchronous mode, the 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.

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

The synchronous mode has an integer sub-mode that is optimized for lowest phase noise integer division. An FOD can be configured for synchronous integer division by writing setting fod_integer_mode = 0x1 (see Table 1). In this mode the FOD divide ratio is programmed using the fod_div_integer register field as indicated in the following equation.

$$\text{FOD divide ratio} = (\text{fod\_div\_integer}[48:40])$$

An FOD in synchronous mode will faithfully follow frequency steering applied to the VCO by the DPLL, or by external software via the write_phase or write_freq register field, or by steering the device frequency reference (e.g., a VCXO overdriving the XIN pin).

1.5 Divider Synchronization

The IODs, FODs, and DPLL feedback divider are automatically synchronized after the device is configured on startup, and 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 DPLL mode.

1.6 DPLL Reference Selection

The DPLL can lock to any of the clock input references Ref 0, Ref 1, Ref 2, or Ref 3. The reference selection can be manual or automatic as determined the dpll_ref_sel_mode register field.

1.6.1 Manual Reference selection

For manual reference selection, the 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 2 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 is given a value of 0.

Table 2. Reference Selection by Pin

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

1.6.2 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 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.

1.6.3 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. 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 the output clocks.

1.6.4 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 3).

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 6. 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.

1.6.5 DPLL External Feedback

In some applications it is useful to use an external feedback path for the 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 setting dpll_fb_div_dis = 0x1 to disable the DPLL feedback divider.

1.7 Reference and Crystal Monitors

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

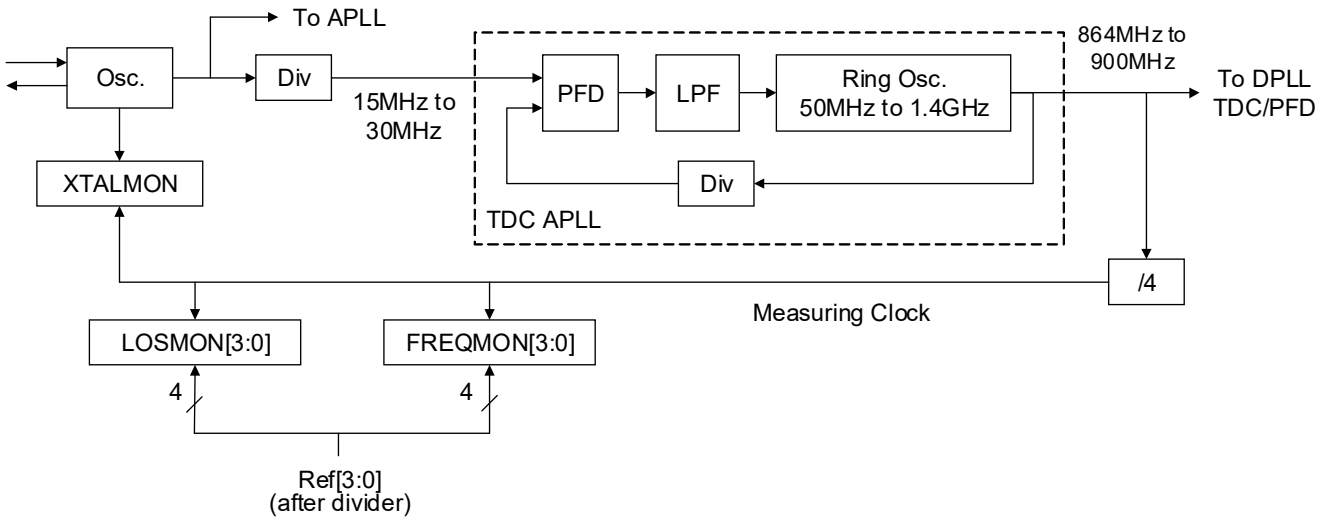


Figure 4. Clock Monitors and TDC APLL

1.8 Digital PLL

The DPLL is 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. The DPLL can lock to reference frequencies from 1kHz to 33MHz; clock inputs with frequencies above 33MHz must be divided using the internal reference dividers. The DPLL steers the APLL using digital frequency control words (FCW) via the Combo Bus.

The DPLL FCWs are relative to the device frequency reference; when the FCW is zero the VCO FFO vs. the frequency reference is zero. The FCW is 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 5). The integral term represents a time average of the VCO FFO. The proportional term represents the short-term changes of the VCO FFO 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 5.

The xtal_trim register field allows host software to apply an FFO to the device frequency reference. The xtal_trim field can be used to improve the frequency accuracy in the Freerun State by compensating for a known FFO of the frequency reference; note that the reference monitors are not affected by this compensation.

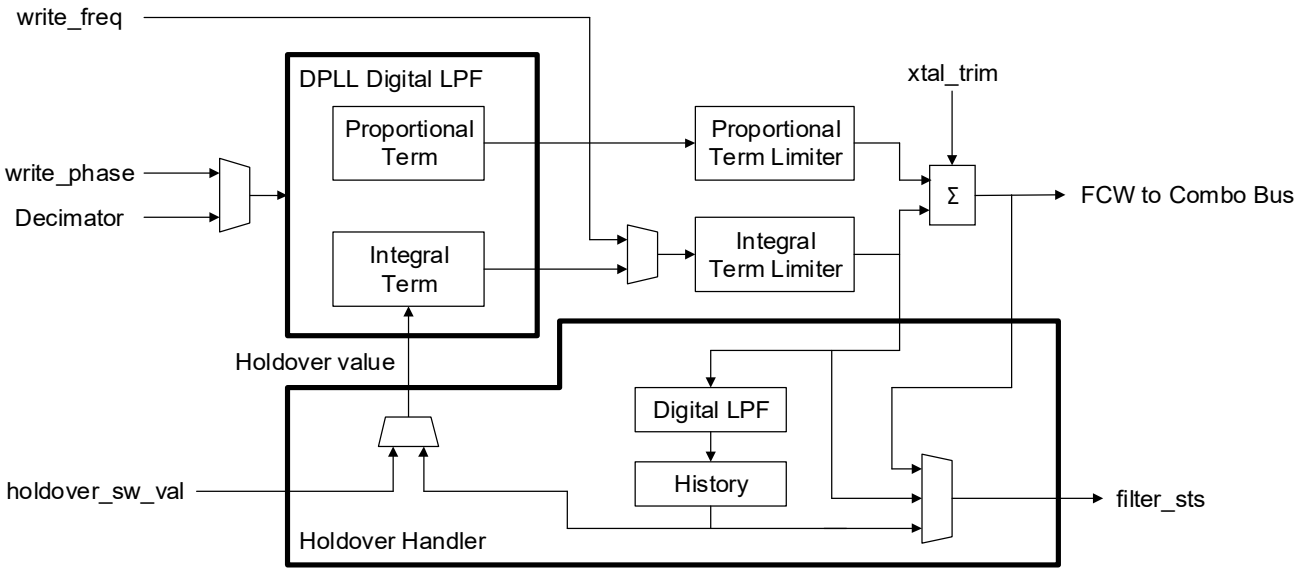


Figure 5. DPLL Digital Low-pass Filter, Limiters and Holdover Handler

1.8.1 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 5).

The Holdover Handler filters the DPLL integrator term with a bypassable 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.

1.8.2 DPLL State Machine

The DPLL can operate in any of six states: Freerun 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 3. 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 3. DPLL States and Modes

Manual State Machine or Automatic State Machine	Forced State or Mode	dpll_en	dpll_mode
Manual See Figure 7	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 6	Automatic Mode		0x6

1.8.3 Freerun State

In the Freerun 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 device frequency reference, plus xtal_trim if used. During the Master Reset Sequence or the Soft Reset Sequence the DPLL will be in the Freerun state (see Figure 6 and Figure 7). There are three combinations of the dpll_en register bit and the dpll_mode register field that will force the DPLL into the Freerun state (see Table 3).

The DPLL can automatically enter the Freerun 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 Freerun state according to the logic shown in Figure 7. When the DPLL state machine is in the automatic mode, the DPLL will automatically enter the Freerun state according to the logic shown in Figure 6. See Table 3 to configure the operating mode of the DPLL state machine.

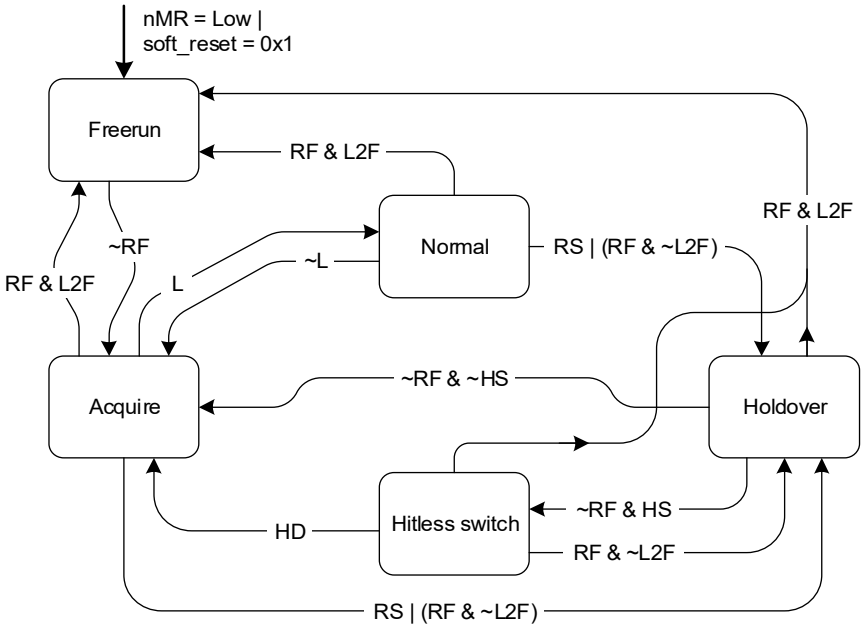
1.8.4 Normal State

The Normal state supports DPLL operation, the DPLL control loop is active and it steers the APLL. While the DPLL is locked, OUTx clocks sourced from the VCO or from FODs in the Synchronous mode 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 3).

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 6. See Table 3 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. 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

Thin lines indicate automatic state transitions
Thick lines indicate register or pin status

Figure 6. DPLL State Machine in Automatic Mode

1.8.5 Write Phase Mode

The Write Phase mode supports DCO operation. It allows host software to steer the FFO of the APLL 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 output of the PFD, via the decimator, is replaced by values from the write_phase register field (see Figure 3 and Figure 5).

1.8.6 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 3).

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 7. 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 6. See Table 3 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 5.

1.8.7 Write Frequency State

The Write Frequency state supports DCO operation, it allows host software to steer the FFO of the APLL using the write_freq register field. The DPLL can be forced into the Write Frequency state by the following settings: dpll_en = 0x1and dpll_mode = 0x3 (see Table 3).

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 5).

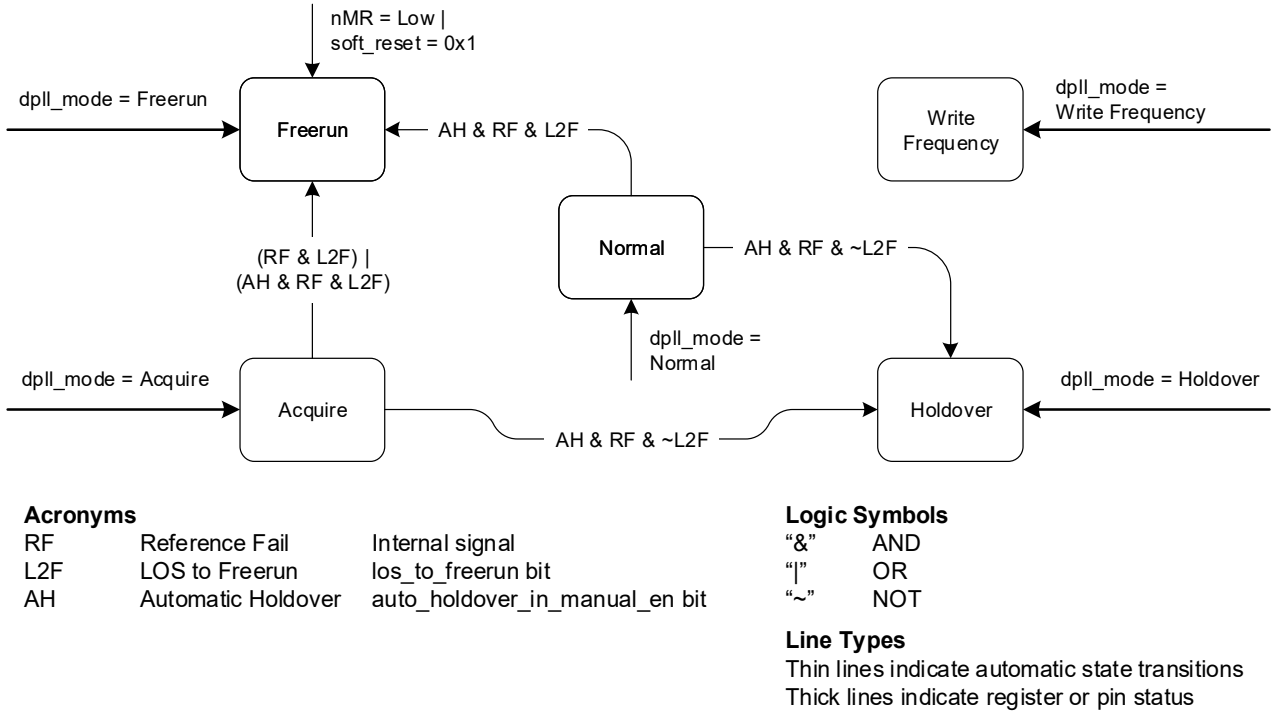


Figure 7. DPLL State Machine in Manual Mode

1.8.8 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 6 and Figure 7). The DPLL can be forced into the Acquire state by the following settings: dpll_en = 0x1and dpll_mode = 0x4 (see Table 3).

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 6. See Table 3 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.

1.8.9 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 6. See Table 3 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 6).

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 7, 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 6). 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.

1.9 Clock Output Paths

The RC32312/RC32308 has two types of clock output path. The paths for OUT[7:4] have access to the VCO/2 clock only. The paths for OUT[3:0] and OUT[11:8] have access to the VCO/2 clock and the FOD[2:0] clocks. Each clock output path includes one IOD and one clock output buffer.

1.9.1 Integer Output Dividers

The IODs are configured using registers in the IOD section.

The 21-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.

1.9.1.1 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_ph_adj_now 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 writing the iod_ph_adj_post_sync bit to 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 is set to 0x0 or 0x1).

1.9.2 Clock Output Path for OUT[7:4]

The clock source selection for OUT[7:4] is illustrated in Figure 8.

Unused clock paths can be powered down by configuring the respective register fields as follows: iod_apll_vco_fanout_en = 0x0; iod_enable = 0x0; out_en_bias = 0x0; out_dis_state = 0x3; and out_driver_en = 0x0.

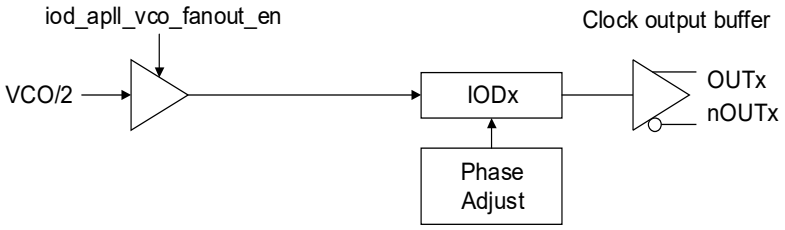


Figure 8. Clock Path for OUT[7:4]

1.9.3 Output Clock Path for OUT[3:0] and OUT[11:8]

The clock source selection for OUT[3:0] and OUT[11:8] is illustrated in Figure 9.

When VCO/2 is not the selected source for an IOD then the respective iod_apll_vco_fanout_en bit should be set to 0x0 to minimize power consumption.

Unused output clock paths can be powered down by configuring the respective register fields as follows: iod_apll_vco_fanout_en = 0x0; iod_mux_sel = 0x7; iod_enable = 0x0; out_en_bias = 0x0; out_dis_state = 0x3; and out_driver_en = 0x0.

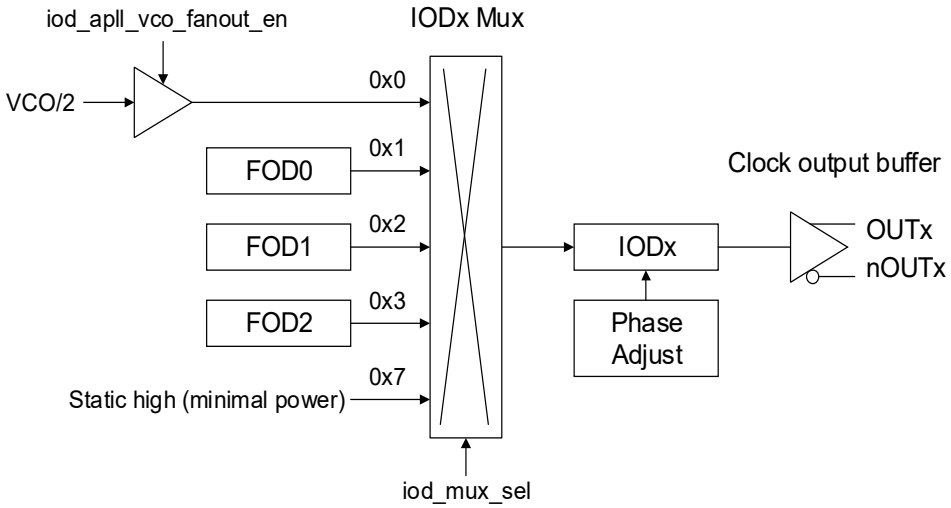


Figure 9. Clock Path for OUT[3:0] and OUT[11:8]

1.9.4 Clock Output Buffers

The clock output buffers are configured using registers in the OUTBUF section.

Unused clock output buffers can be powered down by configuring the respective register fields as follows: out_en_bias = 0x0; out_dis_state = 0x3; and out_driver_en = 0x0.

The HCSL output voltage swing, LVDS output voltage swing, and LVDS common mode voltage are controlled using the out_cnf_hcsl_swing, out_cnf_lvds_amp, and out_lvds_cm_voltage register fields, respectively. The

descriptions for these register fields provide nominal voltage values. For the actual range of voltages associated with each setting, see the device datasheet.

For HCSL outputs (out_mode = 0x0), a boost mode can be enabled for outputs that are configured for 950mV amplitude swing (out_cnf_hcsl_swing = 0xF) by setting the respective out_spare register bit to 0x1. Boost mode increases the HCSL amplitude swing by approximately 10%.

1.10 Status and Control

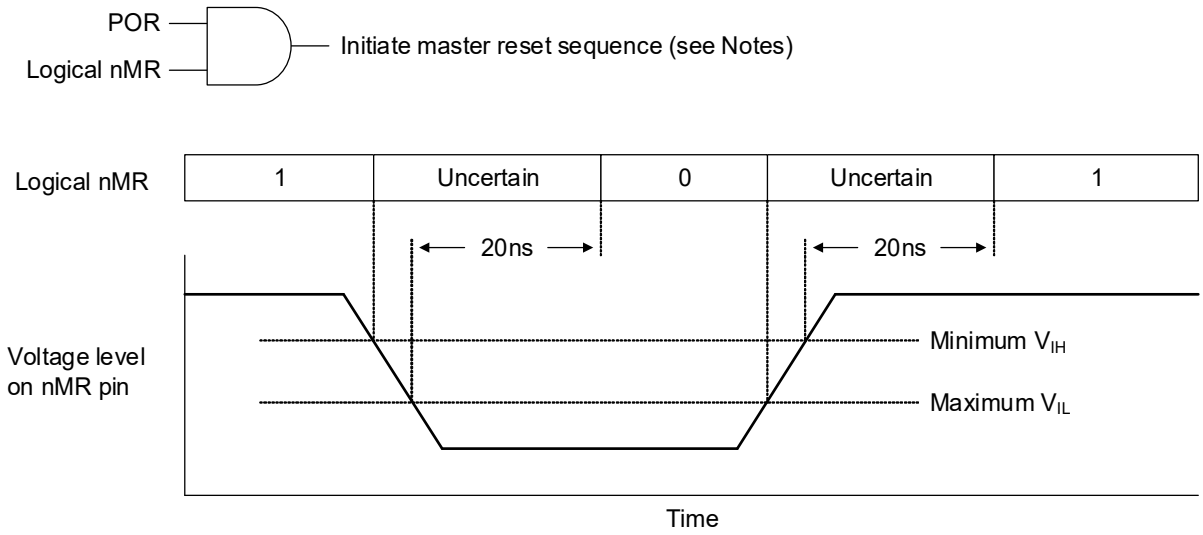
All control and status registers (CSR) are accessed through a 1MHz I^2C or 20MHz SPI slave microprocessor interface. The device can automatically load a configuration from internal one time programmable (OTP) memory. Alternatively, the I^2C master interface can automatically load a configuration from an external EEPROM after reset.

1.11 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 the later of V_{DD_VCO} or V_{DDD33_DIA} then a soft reset or a master reset must be initiated to ensure the input dividers and output dividers are synchronized. 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 5ms after both the V_{DDXO_DCD} and V_{DDD33_DIA} supplies 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 10). 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}.



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 10. 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.

During the master reset sequence all clock outputs are optionally disabled, depending on the value of the out_startup register field. Disabled outputs behave according to the associated out_dis_state register field.

The serial ports are accessible when the device_ready_sts register bit is set to 0x1. Any GPIO can be configured to indicate the state of the device_ready_sts register bit by setting the associated gpio_func register field to 0x18. When a reset sequence completes the rst_done_sts register bit is set to 0x1.

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 register bit is set to 0x1.

1.11.1 Master Reset Sequence

The device can be configured by the config_sel register field 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 register field 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 is initiated until after the configurations have been loaded (i.e., device_ready_sts = 0x1).

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 configurations from OTP memory and/or EEPROM (if present).
   a. Set device_ready_sts 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 machine (see DPLL State Machine and Table 3).
8. Set rst_done_sts to 0x1.

1.11.2 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 relatch_inputs register bit can be configured to update the I^2C address during the soft reset sequence by re-latching the voltage levels as described under Master Reset Sequence. For pins used to update the I^2C address, the voltage levels externally applied must not change from the time a soft reset is initiated until after the soft reset sequence is completed as indicated by rst_done_sts = 0x1.

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

1. Set rst_done_sts 0x0.
2. If relatch_inputs = 0x1.
   a. Latch the levels on the following pins: GPIOs, nCS_A0, SDO_A1, SDA_SDO, and SCL_SCLK.
3. If soft_reset_sel = 0x1.
   a. Synchronize all dividers.
   b. Start the DPLL state machine (see DPLL State Machine and Table 3).
   c. Set rst_done_sts to 0x1.

4. If soft_reset_sel = 0x0.
   a. Calibrate the VCO.
   b. Lock the APLL.
   c. Calibrate the digitally controlled delays (DCD).
   d. Synchronize all dividers.
   e. Start the DPLL state machine (see DPLL State Machine and Table 3).
   f. Set rst_done_sts to 0x1.

The soft reset sequence can be executed starting at Step 4a (Calibrate the VCO) without initiating a soft reset by writing 0x1 to the apll_reinit register bit. The reset sequence can also be executed starting at Step 4d (Synchronize all dividers) without initiating a soft reset by writing 0x1 to the divider_sync register bit. In both of these cases the soft_reset_sel register bit is ignored.

2. 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.

2.1 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 11 shows how page register and offset bytes from each serial transaction interact to address a register within the RC32312, RC32308.

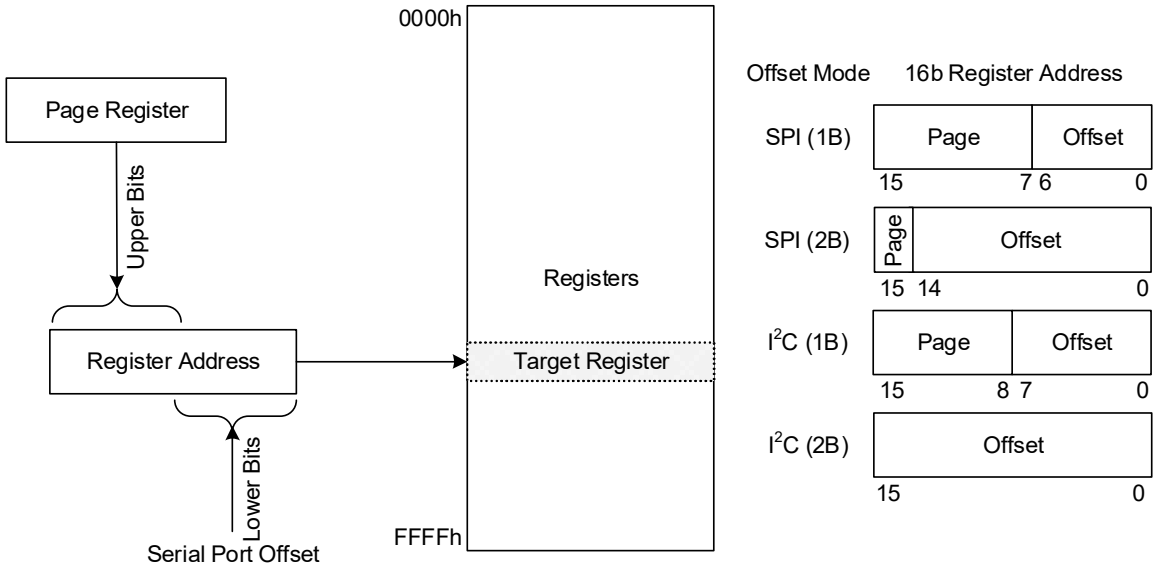


Figure 11. Register Addressing Modes Using Serial Port

2.2 I²C Slave

The I²C slave protocol of the RC32312, RC32308 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 12 shows the sequence of states on the I²C SDA signal for the supported modes of operation.

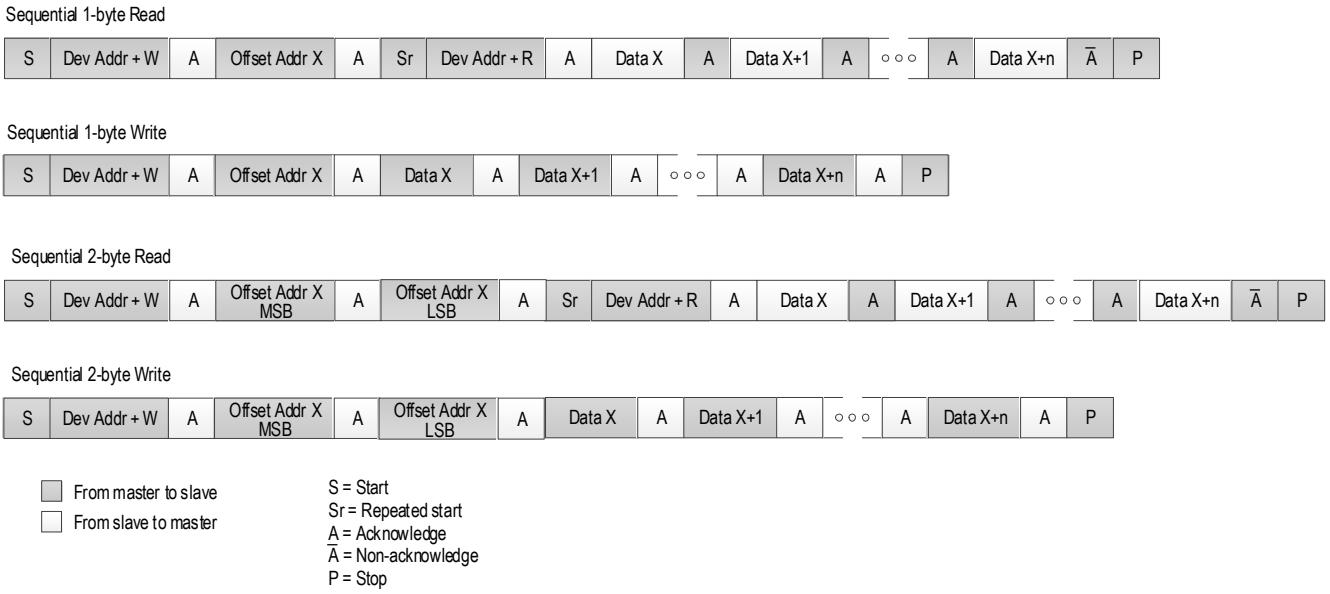


Figure 12. I²C Slave Sequencing

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 if not programmed using the OTP load, or controlled through the nCS_A0, SDO_A1 or GPIO pins.

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 11). 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 RC32312, RC32308 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.

2.2.1 I²C 1-byte (1B) Addressing Example

RC32312, RC32308 I²C 7-bit I²C address 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.
```

2.2.2 I²C 2-byte (2B) Addressing Example

RC32312, RC32308 I²C 7-bit I²C address 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.
```

2.3 SPI Slave

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

The RC32312, RC32308 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 RC32312, RC32308) and data out signals (SDI and SDO respectively). In 3-wire mode, the SDIO signal is used as a single, bidirectional data signal.

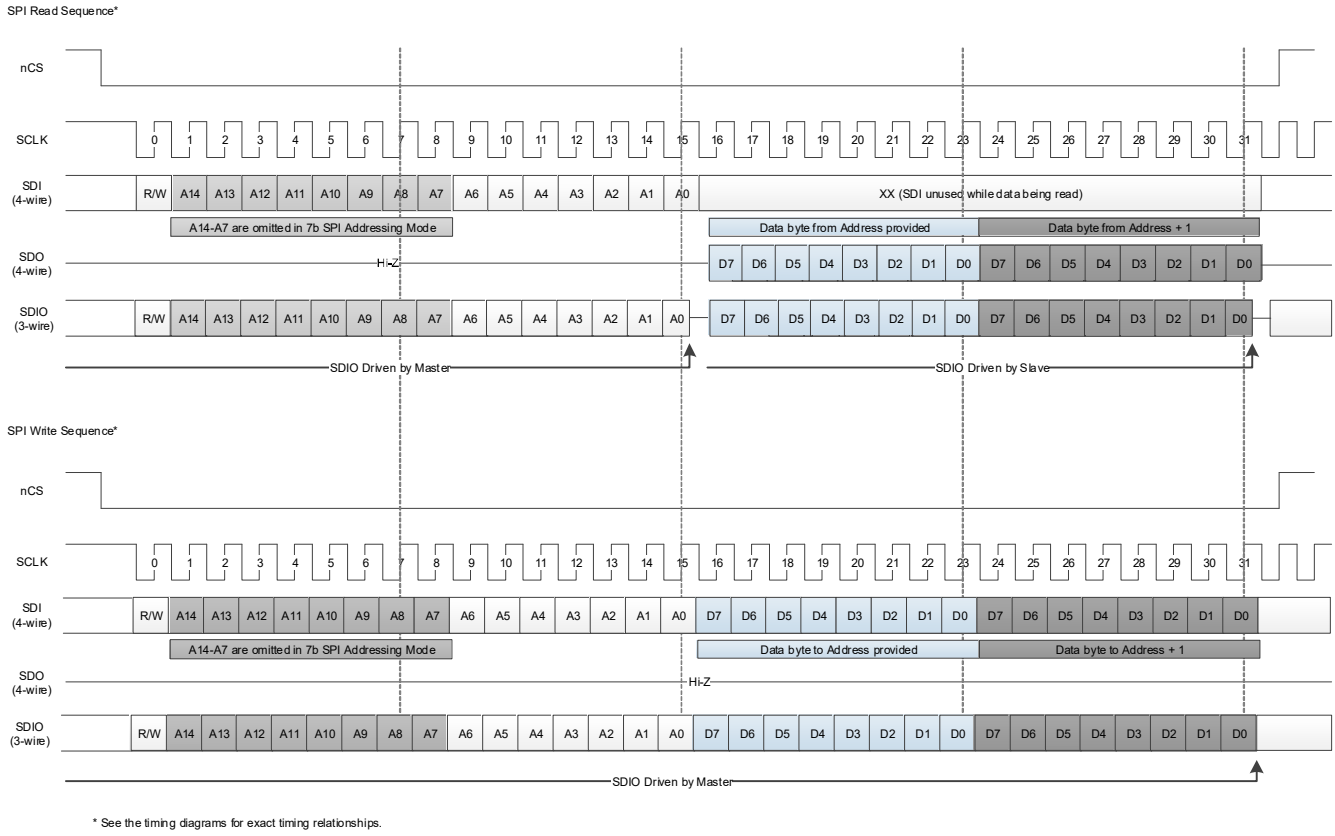


Figure 13. SPI Sequencing

Figure 13 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 11)

- 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 *RC32312, RC32308 Datasheet*.

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

2.3.1 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
```

2.3.2 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
```

3. Registers

Table 4. Register 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
0x70	INPUTBUF[0]	Input Buffer Registers	INPUTBUF
0x72	INPUTBUF[1]	Same as INPUTBUF[0]	INPUTBUF
0x74	INPUTBUF[2]	Same as INPUTBUF[0]	INPUTBUF
0x76	INPUTBUF[3]	Same as INPUTBUF[0]	INPUTBUF
0x80	INPUTMUX	Input Mux Registers	INPUTMUX
0x84	INPUTDIV[0]	Input Buffer Registers	INPUTDIV
0x88	INPUTDIV[1]	Same as INPUTDIV[0]	INPUTDIV
0x8C	INPUTDIV[2]	Same as INPUTDIV[0]	INPUTDIV
0x90	INPUTDIV[3]	Same as INPUTDIV[0]	INPUTDIV
0x94	SYSDIV	System Clock Divider Registers	SYSDIV
0xA0	GPIO[0]	General Purpose IO Registers Instances 0-7 apply to GPIO0 to GPIO9 Instance 8 applies to LOCK	GPIO
0xA8	GPIO[1]	Same as GPIO[0]	GPIO
0xB0	GPIO[2]	Same as GPIO[0]	GPIO
0xB8	GPIO[3]	Same as GPIO[0]	GPIO
0xC0	GPIO[4]	Same as GPIO[0]	GPIO
0xC8	GPIO[5]	Same as GPIO[0]	GPIO
0xD0	GPIO[6]	Same as GPIO[0]	GPIO
0xD8	GPIO[7]	Same as GPIO[0]	GPIO
0xE0	GPIO[8]	Same as GPIO[0]	GPIO
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

Table 4. Register Index

Module Base Address (Hex)	Name	Module Description	Link
0x158	OUTBUF[11]	Same as OUTBUF[0]	OUTBUF
0x160	TDCAPLL	TDC APLL Registers	TDCAPLL
0x170	XTALMON	XTAL Monitor Registers (xtalmon[0] = XIN, xtalmon[1]=REF4)	XTALMON
0x180	LOSMON[0]	LOS Monitor Registers	LOSMON
0x190	LOSMON[1]	Same as LOSMON[0]	LOSMON
0x1A0	LOSMON[2]	Same as LOSMON[0]	LOSMON
0x1B0	LOSMON[3]	Same as LOSMON[0]	LOSMON
0x1C0	FREQMON[0]	Frequency Monitor Registers	FREQMON
0x1E0	FREQMON[1]	Same as FREQMON[0]	FREQMON
0x200	FREQMON[2]	Same as FREQMON[0]	FREQMON
0x220	FREQMON[3]	Same as FREQMON[0]	FREQMON
0x240	APLL	APLL Registers	APLL
0x2A0	IOD[0]	Integer Output Divider Registers	IOD
0x2B0	IOD[1]	Same as IOD[0]	IOD
0x2C0	IOD[2]	Same as IOD[0]	IOD
0x2D0	IOD[3]	Same as IOD[0]	IOD
0x2E0	IOD[4]	Same as IOD[0]	IOD
0x2F0	IOD[5]	Same as IOD[0]	IOD
0x300	IOD[6]	Same as IOD[0]	IOD
0x310	IOD[7]	Same as IOD[0]	IOD
0x320	IOD[8]	Same as IOD[0]	IOD
0x330	IOD[9]	Same as IOD[0]	IOD
0x340	IOD[10]	Same as IOD[0]	IOD
0x350	IOD[11]	Same as IOD[0]	IOD
0x400	FOD[0]	Fractional Output Divider Registers	FOD
0x440	FOD[1]	Same as FOD[0]	FOD
0x480	FOD[2]	Same as FOD[0]	FOD
0x500	DPLL	DPLL Registers	DPLL
0x600	EEPROM	EEPROM Registers	EEPROM

3.1 GLOBAL

Global Control and Status Registers.

Table 5. 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
0x8	DEVICE_CNFG	Device Configuration
0xC	MISC_CNFG	Reset Configuration
0x10	SCRATCH_CNFG	Scratch register
0x14	MISC_CTRL	Reset Control
0x20	STARTUP_STS	Startup status
0x24	DEVICE_STS	Device status

3.1.1 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.

3.1.2 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.

3.1.3 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	0x5	Font ID. Font ID to distinguish die variants. Decode as follows: 0x0 = Font 0 0x2 = Font 1 0x3 = Font 2 0x4 = Font 3 0x5 = Font 4
7:0	reserved	RO	0x53	Reserved

3.1.4 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

3.1.5 DEVICE_CNFG

Device Configuration.

DEVICE_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:8	device_configuration	RW	0x0	Device configuration. This field is unused in test vehicle.
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

DEVICE_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
i2c_addr_sel (continued)				0xA = i2c_addr[2], SDO, nCS 0xB = i2c_addr[2], SDO, gpio[0] 0xC = i2c_addr[2], gpio[1], gpio[0] 0xD = SDO, gpio[1], gpio[0] 0xE = nCS, gpio[1], gpio[0] 0xF = gpio[2], gpio[1], gpio[0]
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

3.1.6 MISC_CNFG

Reset Configuration.

MISC_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7	reserved	RO	0x0	Reserved
6:4	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

MISC_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
3:2	reserved	RO	0x0	Reserved
1:0	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: Note: This setting can cause the reset controller to remain waiting for APLL or DPLL lock and never reach its done state.

3.1.7 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.

3.1.8 MISC_CTRL

Reset Control.

MISC_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7	relatch_inputs	RW	0x0	Relatch inputs on soft reset. 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
6:5	reserved	RO	0x0	Reserved
4	soft_reset_sel	RW	0x0	Soft reset selection. This bit will control whether the soft_reset bit will restart the reset controller from the VCO_CAL state or from the DIVIDER_SYNC state. From the DIVIDER_SYNC, the VCO will not be recalibrated and the APLL will not be relocked. 0x0 = Soft reset recalibrates the VCO, relocks the APLL, recalibrates the DCDs, and resyncs all dividers 0x1 = Resyncs all dividers
3	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

MISC_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
2	apll_reinit	RW	0x0	APLL Reinitialization. Writing this bit to 1 re-starts the startup sequence from the VCO calibration step, including divider synchronization. This bit will have to be re-written to a '0' then a '1' to reissue it.
1	divider_sync	RW	0x0	Divider Synchronization. Write '1' to trigger synchronization of all dividers. This bit will have to be re-written to a '0' then a '1' to reissue it.
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.

3.1.9 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 when a master reset sequence is initiated; or when a soft reset is initiated and relatch_inputs = 0x1. 0x0 = Low 0x1 = High
14	sda_sdio_at_startup_sts	RO	0x0	SDA_SDIO Value Latched at Startup. Value latched when a master reset sequence is initiated; or when a soft reset is initiated and relatch_inputs = 0x1. 0x0 = Low 0x1 = High
13	sdo_a1_at_startup_sts	RO	0x0	SDO_A1 Value Latched at Startup. Value latched when a master reset sequence is initiated; or when a soft reset is initiated and relatch_inputs = 0x1. 0x0 = Low 0x1 = High
12	ncs_a0_at_startup_sts	RO	0x0	NCS_A0 Value Latched at Startup. Value latched when a master reset sequence is initiated; or when a soft reset is initiated and relatch_inputs = 0x1. 0x0 = Low 0x1 = High
11:9	reserved	RO	0x0	Reserved
8	lock_at_startup_sts	RO	0x0	LOCK Value Latched at Startup. Value latched when a master reset sequence is initiated; or when a soft reset is initiated and relatch_inputs = 0x1. 0x0 = Low 0x1 = High
7:0	gpio_at_startup_sts	RO	0x0	GPIOx Value Latched at Startup. Value latched when a master reset sequence is initiated; or when a soft reset is initiated and relatch_inputs = 0x1. 0x0 = Low 0x1 = High

3.1.10 DEVICE_STS

Device status.

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

3.2 INT

Interrupt Registers.

Table 6. 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

3.2.1 INT_EN_CTRL

Interrupt Enable Configuration.

INT_EN_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:39	reserved	RO	0x0	Reserved
38	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
37	reserved	RO	0x0	Reserved
36	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
35	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

INT_EN_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
34	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
33	otp_manual_rdy_int_en	RW	0x0	OTP Manual Request Ready Interrupt Enable. When this field is set to 1, the otp_manual_rdy_int_sts bit contributes to the device interrupt
32	reserved	RO	0x0	Reserved
31	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
30	los3_lmt_int_en	RW	0x0	CLKIN3 Monitor LOS Threshold Exceeded interrupt enable. When this field is set to 1, the los3_lmt_int_sts bit contributes to the device interrupt
29	los2_lmt_int_en	RW	0x0	CLKIN2 Monitor LOS Threshold Exceeded interrupt enable. When this field is set to 1, the los2_lmt_int_sts bit contributes to the device interrupt
28	los1_lmt_int_en	RW	0x0	CLKIN1 Monitor LOS Threshold Exceeded interrupt enable. When this field is set to 1, the los1_lmt_int_sts bit contributes to the device interrupt
27	los0_lmt_int_en	RW	0x0	CLKIN0 Monitor LOS Threshold Exceeded interrupt enable. When this field is set to 1, the los0_lmt_int_sts bit contributes to the device interrupt
26	reserved	RO	0x0	Reserved
25	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
24	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
23	freq3_update_int_en	RW	0x0	CLKIN3 Frequency Monitor Offset Valid interrupt enable. When this field is set to 1, the freq3_update_int_sts bit contributes to the device interrupt
22	freq2_update_int_en	RW	0x0	CLKIN2 Frequency Monitor Offset Valid interrupt enable. When this field is set to 1, the freq2_update_int_sts bit contributes to the device interrupt
21	freq1_update_int_en	RW	0x0	CLKIN1 Frequency Monitor Offset Valid interrupt enable. When this field is set to 1, the freq1_update_int_sts bit contributes to the device interrupt
20	freq0_update_int_en	RW	0x0	CLKIN0 Frequency Monitor Offset Valid interrupt enable. When this field is set to 1, the freq0_update_int_sts bit contributes to the device interrupt.
19	freq3_int_en	RW	0x0	CLKIN3 Frequency Monitor interrupt enable. When this field is set to 1, the freq3_int_sts bit contributes to the device interrupt
18	freq2_int_en	RW	0x0	CLKIN2 Frequency Monitor interrupt enable. When this field is set to 1, the freq2_int_sts bit contributes to the device interrupt
17	freq1_int_en	RW	0x0	CLKIN1 Frequency Monitor interrupt enable. When this field is set to 1, the freq1_int_sts bit contributes to the device interrupt
16	freq0_int_en	RW	0x0	CLKIN0 Frequency Monitor interrupt enable. When this field is set to 1, the freq0_int_sts bit contributes to the device interrupt.
15	reserved	RO	0x0	Reserved
14	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
13	los3_int_en	RW	0x0	CLKIN3 Monitor Loss-of-Signal interrupt enable. When this field is set to 1, the los3_int_sts bit contributes to the device interrupt
12	los2_int_en	RW	0x0	CLKIN2 Monitor Loss-of-Signal interrupt enable. When this field is set to 1, the los2_int_sts bit contributes to the device interrupt

INT_EN_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
11	los1_int_en	RW	0x0	CLKIN1 Monitor Loss-of-Signal interrupt enable. When this field is set to 1, the los1_int_sts bit contributes to the device interrupt
10	los0_int_en	RW	0x0	CLKIN0 Monitor Loss-of-Signal interrupt enable. When this field is set to 1, the los0_int_sts bit contributes to the device interrupt.
9	reserved	RO	0x0	Reserved
8	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.
7	reserved	RO	0x0	Reserved
6	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.
5	reserved	RO	0x0	Reserved
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.

3.2.2 INT_STS

Interrupt Status.

INT_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:39	reserved	RO	0x0	Reserved
38	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_err_evt.
37	reserved	RO	0x0	Reserved
36	dpll_bw_sel_int_sts	RO	0x0	Manual bandwidth select Interrupt Status. When set,indicates that the DPLL bw_sel has changed. This bit reflects the dpll_bw_sel_ch_evt.
35	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 and eeprom_load_fail event bits
34	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 and eeprom_crc_err event bits
33	otp_manual_rdy_int_sts	RO	0x0	OTP Manual Request Ready interrupt status. When set, indicates that the OTP manual command is done. This bit reflects the otp_manual_rdy_evt bit.
32	reserved	RO	0x0	Reserved
31	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 xtalmon_los_evt.

INT_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
30	los3_lmt_int_sts	RO	0x0	CLKIN3 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_evt.
29	los2_lmt_int_sts	RO	0x0	CLKIN2 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_evt.
28	los1_lmt_int_sts	RO	0x0	CLKIN1 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_evt.
27	los0_lmt_int_sts	RO	0x0	CLKIN0 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_evt.
26	reserved	RO	0x0	Reserved
25	dpll_lol_lmt_int_sts	RO	0x0	DPLL Loss-of-lock Threshold Exceeded interrupt status. When set, indicates that the number of the DPLL lost lock exceeded the programmed limit. This bit reflects the dpll_lol_lmt_evt.
24	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.
23	freq3_update_int_sts	RO	0x0	CLKIN3 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.
22	freq2_update_int_sts	RO	0x0	CLKIN2 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.
21	freq1_update_int_sts	RO	0x0	CLKIN1 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.
20	freq0_update_int_sts	RO	0x0	CLKIN0 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.
19	freq3_int_sts	RO	0x0	CLKIN3 Frequency Monitor interrupt status. When set, indicates the FREQMON0 declared freq_fail at some point. This bit reflects the freqmon0_freq_fail_evt bit.
18	freq2_int_sts	RO	0x0	CLKIN2 Frequency Monitor interrupt status. When set, indicates the FREQMON2 declared freq_fail at some point. This bit reflects the freqmon2_freq_fail_evt bit.
17	freq1_int_sts	RO	0x0	CLKIN1 Frequency Monitor interrupt status. When set, indicates the FREQMON1 declared freq_fail at some point. This bit reflects the freqmon1_freq_fail_evt bit.
16	freq0_int_sts	RO	0x0	CLKIN0 Frequency Monitor interrupt status. When set, indicates the FREQMON0 declared freq_fail at some point. This bit reflects the freqmon0_freq_fail_evt bit.
15	reserved	RO	0x0	Reserved
14	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 xtalmon_los_evt bit.

INT_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
13	los3_int_sts	RO	0x0	CLKIN3 Monitor Loss-of-Signal interrupt status. When set, indicates the LOSMON3 declared LOS at some point. This bit reflects the losmon3_los_evt bit.
12	los2_int_sts	RO	0x0	CLKIN2 Monitor Loss-of-Signal interrupt status. When set, indicates the LOSMON2 declared LOS at some point. This bit reflects the losmon2_los_evt bit.
11	los1_int_sts	RO	0x0	CLKIN1 Monitor Loss-of-Signal interrupt status. When set, indicates the LOSMON0 declared LOS at some point. This bit reflects the losmon0_los_evt bit.
10	los0_int_sts	RO	0x0	CLKIN0 Monitor Loss-of-Signal interrupt status. When set, indicates the LOSMON0 declared LOS at some point. This bit reflects the losmon0_los_evt bit.
9	reserved	RO	0x0	Reserved
8	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.
7	reserved	RO	0x0	Reserved
6	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.
5	reserved	RO	0x0	Reserved
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.

3.3 SSI

Slave Serial Interface Registers.

Table 7. 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

Table 7. SSI Register Index (Cont.)

Offset (Hex)	Register Module Base Address: 0x50	
	Register Name	Register Description
0x9	I2C_EVENT	I2C CRC Error Count
0xA	I2C_CRC_ERR_CNT_EVENT	I2C CRC Error Count

3.3.1 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.

3.3.2 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.

3.3.3 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.

3.3.4 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

3.3.5 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	0x0	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	0x1	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

3.3.6 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	0x5	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 = CMOS Output mode and power supply of 3.3V. 0x5 = RESERVED 0x6 = CMOS Output mode and power supply of 1.8V. 0x7 = Reserved

3.3.7 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	0x3	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 = CMOS Output mode and power supply of 3.3V. 0x5 = RESERVED 0x6 = CMOS Output mode and power supply of 1.8V. 0x7 = Reserved

3.3.8 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	0x3	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 = CMOS Output mode and power supply of 3.3V. 0x5 = RESERVED 0x6 = CMOS Output mode and power supply of 1.8V. 0x7 = Reserved

3.3.9 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	0x5	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 = CMOS Output mode and power supply of 3.3V. 0x5 = RESERVED 0x6 = CMOS Output mode and power supply of 1.8V. 0x7 = Reserved

3.3.10 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.

3.3.11 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.

3.4 XO

Crystal Oscillator, Input Buffer and Reference Select Registers.

Table 8. XO Register Index

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

3.4.1 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
11:8	xobuf_digicap_x1	RW	0x0	XIN internal tunning cap select
7:0	reserved	RO	0x3	Reserved

3.5 INPUTBUF

Input Buffer Registers.

Table 9. INPUTBUF Register Index

Offset (Hex)	Register Module Base Address: 0x70	
	Register Name	**Register Description**
0x0	INPUT_CNFG	Input Buffer Configuration

3.5.1 INPUT_CNFG

Input Buffer Configuration.

INPUT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:5	reserved	RO	0x0	Reserved
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)
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

3.6 INPUTMUX

Input Mux Registers.

Table 10. INPUTMUX Register Index

Offset (Hex)	**Register Module Base Address: 0x80**	
	Register Name	**Register Description**
0x0	REF_SEL_CNFG	Reference Mux Configuration

3.6.1 REF_SEL_CNFG

Reference Mux Configuration.

REF_SEL_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
31:12	reserved	RO	0x0	Reserved
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

3.7 INPUTDIV

Input Buffer Registers.

Table 11. INPUTDIV Register Index

Offset (Hex)	Register Module Base Address: 0x84	
	Register Name	Register Description
0x0	INPUT_DIV_CNFG	Input Divider Configuration

3.7.1 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. Input Divider Enable (active low) 0x0 = Enable input divider 0x1 = Disable input divider
22	id_setb	RW	0x0	Input Divider Set. When cleared, the corresponding input divider is held in set mode (bit is active low). Unless this divider is bypassed (id_byp_en = 1), clearing this bit will halt the corresponding input clock. 0x0 = Set 0x1 = Unset
21	reserved	RO	0x0	Reserved
20	id_byp_en	RW	0x1	Input Divider Bypass. Input Clock Pre-divider Bypass Allows the input divider to be bypassed and the input clock input is passed directly to the DPLL. Bypass must be disabled if the input clock frequency is greater than 33 MHz. 0x0 = Divided input clock is selected, divide ratio is id_div_pgm 0x1 = Input clock is selected, effective divide ratio is 1
19:0	id_div_pgm	RW	0x0	Input Divider ratio. The input clock frequency divided by this value must be no more than 33 MHz, and must be equal to the DPLL feedback clock frequency. The minimum divide value is 2. To divide by 1 (when the input clock frequency is no more than 33 MHz), bypass the divider by setting id_byp_en to 1. To support hitless switching between input clocks, they must have the same nominal frequency after the pre-divider.

3.8 SYSDIV

System Clock Divider Registers.

Table 12. SYSDIV Register Index

Offset (Hex)	Register Module Base Address: 0x94	
	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

3.8.1 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.

3.8.2 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	0x3B	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.

3.8.3 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

3.9 GPIO

General Purpose IO Registers.

Instances 0-7 apply to GPIO0 to GPIO9.

Instance 8 applies to LOCK.

Table 13. GPIO Register Index

Offset (Hex)	Register Module Base Address: 0xA0	
	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

3.9.1 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
13	gpio_resync	RW	0x1	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.
12	gpio_pol	RW	0x0	GPIO polarity. Inverts input or output 0x0 = Do not invert 0x1 = Invert
11	reserved	RO	0x0	Reserved

GPIO_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
10:8	gpio_startup_mode	RW	0x1	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 = Reserved 0x6 = Reserved 0x7 = Reserved
7:0	gpio_func	RW	0x20,0x20,0x20,0x20,0x20,0x20,0x20,0x1B	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) 0x8 = REFIN0 LOF (loss of frequency) (output) 0x9 = REFIN1 LOF (loss of frequency) (output) 0xA = REFIN2 LOF (loss of frequency) (output)
gpio_func (continued)				0xB = REFIN3 LOF (loss of frequency) (output) 0xC = REFIN0 ref_invalid_sts (output) 0xD = REFIN1 ref_invalid_sts (output) 0xE = REFIN2 ref_invalid_sts (output) 0xF = REFIN3 ref_invalid_sts (output) 0x11 = XIN LOS limit (output) 0x12 = REFIN0 LOS limit (output) 0x13 = REFIN1 LOS limit (output) 0x14 = REFIN2 LOS limit (output) 0x15 = REFIN3 LOS limit (output)
gpio_func (continued)				0x17 = Load complete (OTP or EEPROM loaded) (output) 0x18 = Device Ready (startup sequence complete) (output) 0x19 = APLL rail low (output) 0x1A = APLL rail high (output) 0x1B = APLL lock (from frequency-based lock detect) (output) 0x1C = APLL lol (output) 0x1D = APLL lol_lmt (output) 0x1E = APLL LF lock (VCTRL-based lock) (output) 0x1F = FCL lock (aux loop status) (output) 0x20 = General purpose input (input)

GPIO_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
	gpio_func (continued)			0x30 = DPLL lock (output) 0x31 = DPLL lol (output) 0x32 = DPLL lol_lmt (output) 0x33 = DPLL state[0] (output) 0x34 = DPLL state[1] (output) 0x35 = DPLL state[2] (output) 0x3C = TOD compare (output) (set gpio_resync to 0) [Reserved] 0x4C = Global OE (input) 0x4D = Group OE[0:3] (input) 0x4E = Group OE[4:7] (input)
	gpio_func (continued)			0x4F = Group OE[8:11] (input) 0x50 = OE[0] (input) 0x51 = OE[1] (input) 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)
	gpio_func (continued)			0x59 = OE[9] (input) 0x5A = OE[10] (input) 0x5B = OE[11] (input) 0x5C = DPLL_force_freerun (input) 0x5D = DPLL_force_holdover (input) 0x5E = DPLL_force_normal (input) 0x5F = DPLL_force_acquire (input) 0x60 = DPLL PBO clear (input) 0x61 = DPLL BW_SEL (input) 0x62 = DPLL REFIN_SEL[0] (input)
	gpio_func (continued)			0x63 = DPLL REFIN_SEL[1] (input) 0x6C = REFIN0_FORCE_LOS (input) 0x6D = REFIN1_FORCE_LOS (input) 0x6E = REFIN2_FORCE_LOS (input) 0x6F = REFIN3_FORCE_LOS (input) 0x7F = Device Global Interrupt (output) 0x80 = Device interrupt[0] - APLL LOL (frequency-based lock detector) (output) 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)

GPIO_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
	gpio_func (continued)			0x84 = Device interrupt[4] - DPLL LOL (output) 0x86 = Device interrupt[6] - DPLL HOLDOVER (output) 0x88 = Device interrupt[8] - DPLL STATE CHANGE (output) 0x8A = Device interrupt[10] - REFIN0 LOS (output) 0x8B = Device interrupt[11] - REFIN1 LOS (output) 0x8C = Device interrupt[12] - REFIN2 LOS (output) 0x8D = Device interrupt[13] - REFIN3 LOS (output) 0x8E = Device interrupt[14] - XIN0 LOS (output) 0x90 = Device interrupt[16] - REFIN0 LOF (output) 0x91 = Device interrupt[17] - REFIN1 LOF (output)
	gpio_func (continued)			0x92 = Device interrupt[18] - REFIN2 LOF (output) 0x93 = Device interrupt[19] - REFIN3 LOF (output) 0x94 = Device interrupt[20] - REFIN0 FREQ UPDATE (output) 0x95 = Device interrupt[21] - REFIN1 FREQ UPDATE (output) 0x96 = Device interrupt[22] - REFIN2 FREQ UPDATE (output) 0x97 = Device interrupt[23] - REFIN3 FREQ UPDATE (output) 0x98 = Device interrupt[24] - APLL LOL LMT (output) 0x99 = Device interrupt[25] - DPLL LOL LMT (output) 0x9B = Device interrupt[27] - REFIN0 LOS LMT (output) 0x9C = Device interrupt[28] - REFIN1 LOS LMT (output)
	gpio_func (continued)			0x9D = Device interrupt[29] - REFIN2 LOS LMT (output) 0x9E = Device interrupt[30] - REFIN3 LOS LMT (output) 0x9F = Device interrupt[31] - XIN0 LOS LMT (output) 0xA1 = Device interrupt[33] - OTP MANUAL RDY (output) 0xA2 = Device interrupt[34] - OTP/EEPROM CRC ERR (output) 0xA3 = Device interrupt[35] - OTP/EEPROM FAIL EVT (output) 0xA4 = Device interrupt[40] - TIME SYNC TOD COMPARE (output) (set gpio_resync to 0) [Reserved] 0xA5 = Device interrupt[41] - TIME SYNC TDC FIFO CNT ALARM (output) [Reserved] 0xA6 = Device interrupt[42] - TIME SYNC TDC FIFO OVERRUN (output) [Reserved] 0xA7 = Device interrupt[36] - DPLL BANDWIDTH SELECT (output)
	gpio_func (continued)			0xA9 = Device interrupt[38] - I2C CRC ERR (output) 0xD4 = clk_cali_div8 (output) (set gpio_resync to 0) 0xD5 = clk_xin (output) (set gpio_resync to 0) 0xD6 = clk_dcd (output) (set gpio_resync to 0) 0xD7 = tdc_clk[0] (output) (set gpio_resync to 0) 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) 0xDB = tdc_clk[4] (output) (set gpio_resync to 0) 0xDE = refin_divided[0] (output) (set gpio_resync to 0)
	gpio_func (continued)			0xDF = refin_divided[1] (output) (set gpio_resync to 0) 0xE0 = refin_divided[2] (output) (set gpio_resync to 0) 0xE1 = refin_divided[3] (output) (set gpio_resync to 0) 0xE3 = clk_apll_test_mux (output) (set gpio_resync to 0)

3.9.2 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

3.9.3 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	0x1,0x1,0x1,0x1,0x1,0x1,0x1,0x1	Output enable 0x0 = Enable 0x1 = Disable
2:0	pad_gpio_drv	RW	0x5	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 = CMOS Output mode and power supply of 3.3V. 0x5 = RESERVED 0x6 = CMOS Output mode and power supply of 1.8V. 0x7 = Reserved

3.9.4 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.

3.10 OUTBUF

Output Buffer Registers.

Table 14. OUTBUF Register Index

Offset (Hex)	Register Module Base Address: 0x100	
	Register Name	**Register Description**
0x0	OUT_CNFG	Output Buffer General Configuration
0x1	OUT_MODE_CNFG	Output Buffer Mode Configuration
0x2	OUT_HCSL_CMOS_CNFG	Output Buffer HCSL CMOS Configuration
0x3	OUT_LVDS_CNFG	Output Buffer LVDS Configuration
0x4	OUT_EN_CTRL	Output Pad Enables
0x6	OUT_SPARE	Spare Output Buffer Bits

3.10.1 OUT_CNFG

Output Buffer General Configuration.

OUT_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:5	reserved	RO	0x0	Reserved
4	out_en_bias	RW	0x1	Output Driver Bias Enable. Output Driver Bias Enable. 0x0 = Disable 0x1 = Enable
3:0	reserved	RW	0x8	Reserved

3.10.2 OUT_MODE_CNFG

Output Buffer Mode Configuration.

OUT_MODE_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:4	reserved	RO	0x0	Reserved
3:2	out_dis_state	RW	0x0	Output Driver disabled state. Controls the state of OUtx/nOUTx when the output driver is disabled by the reset state machine, out_driver_en bit, or by GPIO. 0x0 = Held Low / Low (except LVDS/HCSL mode is held Low / High). 0x1 = Held Low / High. 0x2 = Held Hi-Z / Hi-Z. 0x3 = Output power down (regardless of output enable of the driver)
1:0	out_mode	RW	0x3	Output Driver type. Output Driver type. 0x0 = HCSL 0x1 = Reserved 0x2 = LVDS 0x3 = CMOS

3.10.3 OUT_HCSL_CMOS_CNFG

Output Buffer HCSL CMOS Configuration.

OUT_HCSL_CMOS_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:6	out_cmos_mode	RW	0x1	Output Driver CMOS mode. Controls how OUTx and nOUTx are driven when CMOS mode is selected. 0x0 = OUTx, nOUTx are driven with the same phase. 0x1 = OUTx, nOUTx are driven with the opposite phase. 0x2 = Only OUTx is driven. nOUTx is held low. 0x3 = Only nOUTx is driven. OUTx is held low.
5	reserved	RO	0x0	Reserved
4	out_hcsl_termination_en	RW	0x0	Output Driver HCSL termination enable. Controls the internal HCSL termination. 0x0 = Internal HCSL termination is disabled. An external termination resistor of 50 ohms to ground is required. 0x1 = Internal HCSL termination is enabled, providing an internal 50 ohm resistor to ground.

OUT_HCSL_CMOS_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
3:0	out_cnf_hcsl_swing	RW	0x0	Output Driver HCSL amplitude control. Controls the amplitude of the output driver when CML mode is selected. Each value provides a 50mV increment. 0x0 = 200mV 0x1 = 250mV 0x2 = 300mV 0x3 = 350mV 0x4 = 400mV 0x5 = 450mV 0x6 = 500mV 0x7 = 550mV 0x8 = 600mV 0x9 = 650mV
out_cnf_hcsl_swing (continued)				0xA = 700mV 0xB = 750mV 0xC = 800mV 0xD = 850mV 0xE = 900mV 0xF = 950mV

3.10.4 OUT_LVDS_CNFG

Output Buffer LVDS Configuration.

OUT_LVDS_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:3	reserved	RO	0x0	Reserved
2	out_cnf_lvds_amp	RW	0x0	Output Driver LVDS amplitude control. Controls the amplitude of the output driver when LVDS mode is selected. 0x0 = 350mV 0x1 = 400mV
1:0	out_lvds_cm_voltage	RW	0x0	Output Driver LVDS common mode voltage control. Controls the common mode voltage of the output driver when LVDS mode is selected. 0x0 = RESERVED 0x1 = 800mV 0x2 = 900mV 0x3 = 1000mV

3.10.5 OUT_EN_CTRL

Output Pad Enables.

OUT_EN_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	0x1,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0	Output Pad synchronous enable. Synchronously enables/disables output drivers when oe_source_sel=0x1. After this bit is set, the clock output will be enabled after a delay of up to 3 output clock periods. The clock signal from an output driver enabled from Hi-Z (out_dis_state=0x2) will be chaotic while it reaches steady state. To minimize power consumed by an unused output driver and the corresponding IOD apply the following settings: 1. To power-down the output driver: a) out_en_bias=0x0. b) out_dis_state=0x3. c) out_driver_en=0x0. 2. To power-down IOD[7:4]: a) iod_enable=0x0. b) iod_apll_vco_fanout_en=0x0. 3. To power-down IOD[3:0] & [11:8]: a) iod_enable=0x0. b) iod_apll_vco_fanout_en=0x0. c) iod_mux_sel=0x7. 0x0 = Disable 0x1 = Enable

3.10.6 OUT_SPARE

Spare Output Buffer Bits.

OUT_SPARE Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:0	out_spare	RW	0x0	10% boost for HCSL swing. Note: Available for 950mV swing only. out_spare[0] bits <7:0> correspond to en_hcsl_boost[7:0] out_spare[1] bits <3:0> correspond to en_hcsl_boost[11:8] 0 = Disable HCSL amplitude boost 1 = Enable HCSL amplitude boost

3.11 TDCAPLL

TDC APLL Registers.

Table 15. TDCAPLL Register Index

Offset (Hex)	Register Module Base Address: 0x160	
	Register Name	Register Description
0x2	TDC_FB_DIV_INT_CNFG	TDC Feedback Divider Integer Configuration
0x3	TDC_REF_DIV_CNFG	TDC Reference Divider Configuration
0x9	TDC_ENABLE_CTRL	TDC Filter Status

3.11.1 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.

3.11.2 TDC_REF_DIV_CNFG

TDC Reference Divider Configuration.

TDC_REF_DIV_CNFG Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
7:3	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

3.11.3 TDC_ENABLE_CTRL

TDC Filter Status.

TDC_ENABLE_CTRL Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
7:1	reserved	RO	0x0	Reserved
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.

3.12 XTALMON

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

Table 16. XTALMON Register Index

Offset (Hex)	Register Module Base Address: 0x170	
	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

3.12.1 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.

3.12.2 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).

XTALMON_WINDOW_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:5	reserved	RO	0x0	Reserved
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.

3.12.3 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.

3.12.4 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

3.12.5 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
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

3.12.6 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.

3.12.7 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

3.13 LOSMON

LOS Monitor Registers.

Table 17. LOSMON Register Index

Offset (Hex)	Register Module Base Address: 0x180	
	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

3.13.1 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.

3.13.2 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.
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).

LOSMON_WINDOW_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
8:6	reserved	RO	0x0	Reserved
5	los_fail_mask	RW	0x0	LOS Monitor Failure Mask. Masks the LOS monitor status contribution to ref_invalid_sts. 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.

3.13.3 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.

3.13.4 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

3.13.5 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

3.13.6 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.

3.13.7 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

LOSMON_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
1	ref_invalid_sts	RO	0x1	Reference Clock Invalid status. Indicates whether this reference clock 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	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

3.14 FREQMON

Frequency Monitor Registers.

Table 18. FREQMON Register Index

Offset (Hex)	Register Module Base Address: 0x1C0	
	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

3.14.1 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	reserved	RO	0x0	Reserved
28:0	freq_div_ratio	RW	0x2DC 6C0	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.

3.14.2 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.

3.14.3 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.

3.14.4 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

3.14.5 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

3.14.6 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)

3.15 APLL

APLL Registers.

Table 19. APLL Register Index

Offset (Hex)	Register Module Base Address: 0x240	
	Register Name	**Register Description**
0xF	APLL_DCD_SDM_CNFG	APLL DCD SDM Configuration
0x10	APLL_FB_DIV_FRAC_CNFG	APLL Feedback Divider Fraction
0x18	APLL_FB_DIV_INT_CNFG	APLL Feedback Divider Integer
0x1A	APLL_DCD_CAL_CNFG	APLL DCD Calibration Configuration

3.15.1 APLL_DCD_SDM_CNFG

APLL DCD SDM Configuration.

APLL_DCD_SDM_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:2	reserved	RO	0x0	Reserved
1:0	sdm_order	RW	0x3	DCD SDM order. Order of the SDM for the DCD delay code. 0 = integer, zero order, 1 = 1st order, 2 = 2nd order, 3 = 3rd order.

3.15.2 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	0x1CD555555 5	APLL Feedback Divider Fraction. APLL feedback divider numerator value. The denominator is a fixed value of 2^38.

3.15.3 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	0xCB	APLL Feedback Divider Integer. APLL feedback divider integer value.

3.15.4 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 to 1 when the apll_fb_div_frac=0 AND the DPLL is in freerun mode
13:0	reserved	RO	0x1a24	Reserved

3.16 IOD

Integer Output Divider Registers.

Table 20. IOD Register Index

Offset (Hex)	Register Module Base Address: 0x2A0	
	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
0x7	IOD_EN_CTRL	Integer Output Divider Enable Register
0x8	IOD_PHASE_EN_CTRL	Integer Output Divider Phase Adjust Trigger

3.16.1 IOD_DIV_CNFG

Integer Output Divider Ratio Configuration Register.

IOD_DIV_CNFG Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
31:21	reserved	RO	0x0	Reserved
20:0	iod_divider	RW	0x20	Integer Output Divider Ratio. Integer output divider ratio.

3.16.2 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 2x the VCO period, if selecting an FOD clock, the step size is 1x the selected FOD 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 gets applied when the iod_ph_adj_now gets written to 1, or gets applied immediately after a divider resync event if the iod_ph_adj_post_sync is set to one.

3.16.3 IOD_CNFG

Integer Output Divider Configuration Register.

IOD_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:5	reserved	RO	0x0	Reserved
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	iod_apll_vco_fanout_en	RW	0x1	Enable fanout buffer from APLL to IOD or IOD selection mux. For IOD[7:4], when set, this bit enables the APLL VCO fanout buffer to the IOD. For IOD[3:0] and IOD[11:8], when set, this bit enables the APLL VCO fanout buffer to the IOD selection mux. 0x0 = Disable 0x1 = Enable
2:0	iod_mux_sel	RW	0x0	Integer Output Divider Clock Select. This register field applies to IOD[3:0] and IOD[11:8] only. 0x0 = VCO/2 0x1 = FOD0 0x2 = FOD1 0x3 = FOD2 0x4 = RESERVED 0x5 = RESERVED 0x6 = RESERVED 0x7 = Static high (Minimal power)

3.16.4 IOD_EN_CTRL

Integer Output Divider Enable Register.

IOD_EN_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:1	reserved	RO	0x0	Reserved
0	iod_enable	RW	0x1,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0	IOD Enable. This bit enables or disables the IOD. 0x0 = Disabled 0x1 = Enabled

3.16.5 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).

3.17 FOD

Fractional Output Divider Registers.

Table 21. FOD Register Index

Offset (Hex)	Register Module Base Address: 0x400	
	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_EN_CTRL	FOD enable / power down
0x22	FOD_PHASE_EN_CTRL	FOD Phase Adjust Trigger
0x28	FOD_WRITE_FREQ_CTRL	FOD Write Frequency Control

3.17.1 FOD_CNFG

FOD General Configuration.

FOD_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15:5	reserved	RO	0x0	Reserved
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

FOD_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
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	fod_integer_mode	RW	0x0	FOD integer divide mode. force the FOD to integer divide mode. Fod_div_fraction, _numerator and _denominator are not used. 0x0 = Fractional divide mode. 0x1 = Integer divide mode. FOD is optimized internally for integer divides.

3.17.2 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.

3.17.3 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 for the formula.

3.17.4 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	0x8000 000000	FOD denominator. FOD denominator for Synchronous mode (fod_sync_mode=1) only. See fod_div_integer for the formula.

3.17.5 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 low 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

3.17.6 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

3.17.7 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.

3.17.8 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

3.18 DPLL

DPLL Registers.

Table 22. 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
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
0x30	DPLL_FB_DIV_NUM_CNFG	DPLL Feedback Divider Numerator Configuration
0x38	DPLL_FB_DIV_DEN_CNFG	DPLL Feedback Divider Denominator Configuration

Table 22. DPLL Register Index

Offset (Hex)	Register Module Base Address: 0x500	
	Register Name	Register Description
0x40	DPLL_FB_DIV_INT_CNFG	DPLL Feedback Divider Integer Configuration
0x44	DPLL_FB_CORR_CNFG	DPLL Feedback Divider Correction 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_EVENT	DPLL Events
0x71	DPLL_STS	DPLL Status
0x72	DPLL_LOL_CNT_STS	DPLL Loss of Lock Counter Status
0x74	DPLL_WR_FREQ_PHASE_TIMER_STS	DPLL Timed Write Frequency and Phase Status
0x80	DPLL_FILTER_STS	DPLL Filter Status
0x88	DPLL_PHASE_STS	DPLL Phase Status

3.18.1 DPLL_REF_FB_CNFG

DPLL Reference Configuration.

DPLL_REF_FB_CNFG Bit Field Descriptions				
Bit Field	Field Name	Field Type	Default Value	Description
15	reserved	RO	0x0	Reserved
14:9	dpll_gpio_ref_sel_debounce_delay	RW	0x6	DPLL GPIO reference select debounce delay. Sets the debounce delay in system clock cycles for DPLL reference selection through GPIO.
8:6	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 = RESERVED 0x6 = RESERVED 0x7 = RESERVED
5	dpll_revertive_en	RW	0x0	DPLL revertive reference switch 0x0 = Non-revertive 0x1 = Revertive

DPLL_REF_FB_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
4	dpll_hitless_en	RW	0x0	DPLL hitless reference switch 0x0 = Hitless disabled 0x1 = Hitless enabled
3:2	dpll_ref_sel	RW	0x0	DPLL manual reference clock selection 0x0 = REF 0 0x1 = REF 1 0x2 = REF 2 0x3 = REF 3
1:0	dpll_ref_sel_mode	RW	0x0	DPLL reference clock selection mode 0x0 = Controlled by dpll_ref_sel 0x1 = GPI/GPIO 0x2 = Auto mode 0x3 = Reserved

3.18.2 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

3.18.3 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
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.

DPLL_MODE_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
3	reserved	RO	0x0	Reserved
2:0	dpll_mode	RW	0x6	DPLL mode selection. Selects the mode or state for the DPLL. Selections shown without brackets apply when dpll_en=0x1. Selections shown within brackets apply when dpll_en=0x0. 0x0 = Freerun (Freerun) 0x1 = Normal (Reserved) 0x2 = Holdover (Reserved) 0x3 = Write Frequency (Reserved) 0x4 = Acquire (Reserved) 0x5 = Reserved (Reserved) 0x6 = Automatic (Freerun) 0x7 = Reserved (Reserved)

3.18.4 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.

3.18.5 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.

3.18.6 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	0x12	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	0xB	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.

3.18.7 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 1.1 dB 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 1.1 dB peaking in the frequency domain jitter transfer function.
7:6	reserved	RO	0x0	Reserved
5:3	normal_damping_shift	RW	0x0	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 0.1 dB peaking in the frequency domain jitter transfer function.
2:0	normal_damping_mult	RW	0x0	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 0.1 dB peaking in the frequency domain jitter transfer function.

3.18.8 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**
31:0	phase_slope_limit	RW	0xFFFF FFFF	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.

3.18.9 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.

3.18.10 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. This unsigned quantity limits the magnitude of the filter integral path output. 1 LSB = 2^-20 = 0.954 ppm.

3.18.11 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 bw config and would require a phase reading of the newly selected valid reference.

3.18.12 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.
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.

3.18.13 DPLL_PHASE_OFFSET_CNFG

DPLL Phase Offset Configuration.

DPLL_PHASE_OFFSET_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
127:126	reserved	RO	0x0	Reserved
125:96	ref3_phase_offset	RW	0x0	Phase Offset for ref 3. Manually sets the phase offset between the reference and feedback clocks. This is a 30-bit 2's complement value. The resolution is the TDC resolution / 8 (~ 2.3 ps) and the range is ~ +-1.26 ms. 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 [29:24]) is written, the new value is applied to the DPLL.
95:94	reserved	RO	0x0	Reserved
93:64	ref2_phase_offset	RW	0x0	Phase Offset for ref 2. Manually sets the phase offset between the reference and feedback clocks. This is a 30-bit 2's complement value. The resolution is the TDC resolution / 8 (~ 2.3 ps) and the range is ~ +-1.26 ms. 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 [29:24]) is written, the new value is applied to the DPLL.
63:62	reserved	RO	0x0	Reserved
61:32	ref1_phase_offset	RW	0x0	Phase Offset for ref 1. Manually sets the phase offset between the reference and feedback clocks. This is a 30-bit 2's complement value. The resolution is the TDC resolution / 8 (~ 2.3 ps) and the range is ~ +-1.26 ms. 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 [29:24]) is written, the new value is applied to the DPLL.
31:30	reserved	RO	0x0	Reserved
29:0	ref0_phase_offset	RW	0x0	Phase Offset for ref 0. Manually sets the phase offset between the reference and feedback clocks. This is a 30-bit 2's complement value. The resolution is the TDC resolution / 8 (~ 2.3 ps) and the range is ~ +-1.26 ms. 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 [29:24]) is written, the new value is applied to the DPLL.

3.18.14 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.

3.18.15 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.

3.18.16 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. 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.

3.18.17 DPLL_FB_CORR_CNFG

DPLL Feedback Divider Correction Configuration.

DPLL_FB_CORR_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
15	dpll_fb_div_dis	RW	0x0	DPLL Feedback Divider Disable. Disables the DPLL feedback divider. May be set to 1 to reduce power consumption when an external feedback clock is selected with dpll_fb_sel. The feedback divider is automatically disabled when dpll_en is set to 0. 0x0 = DPLL feedback divider enabled if dpll_en is set to 1 0x1 = DPLL feedback divider disabled
14:7	reserved	RO	0x0	Reserved
6:0	pec_corr_mult	RW	0x0	Feedback Correction Multiplier. Multiplier to get the FB SDM remainder bits on the same resolution as the TDC phase detector bits.(resolution ~18.7 ps if the TDC APLL runs at 864MHz). After device startup, should only be changed while tdc_en is set to 0. The value should be set as follows: Tvco/TDC_step * 128/den[47:41 Where Tvco is the period of the APLL VCO clock, and TDC_step is the TDC resolution (18.7ps typical)

3.18.18 DPLL_LOCK_CNFG

DPLL Lock Configuration.

DPLL_LOCK_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:36	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).
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.

3.18.19 DPLL_CTRL

DPLL Control.

DPLL_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
7:3	reserved	RO	0x0	Reserved
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 = Synthesizer/DCO mode. 0x1 = Jitter Attenuator mode. DPLL is enabled.

3.18.20 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.

3.18.21 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.

3.18.22 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.

3.18.23 DPLL_WR_PHASE_CTRL

DPLL Write Phase Control.

DPLL_WR_PHASE_CTRL Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:33	reserved	RO	0x0	Reserved
32:0	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 / 8

3.18.24 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 this multi-byte register will only take effect when the most significant byte (bits [28:24]) are written, the new value is applied to the DPLL.

3.18.25 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

3.18.26 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.

3.18.27 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

DPLL_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
2:1	dpll_ref_sel_sts	RO	0x0	DPLL reference clock selection status. Indicates the reference clock selected by the DPLL. 0x0 = CLKIN0 0x1 = CLKIN1 0x2 = CLKIN2 0x3 = CLKIN3
0	dpll_lock_sts	RO	0x0	DPLL lock status. Indicates the DPLL lock status: 0x0 = Unlocked 0x1 = Locked

3.18.28 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.

3.18.29 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.

3.18.30 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.

3.18.31 DPLL_PHASE_STS

DPLL Phase Status.

DPLL_PHASE_STS Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
63:33	reserved	RO	0x0	Reserved
32:0	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).

3.19 EEPROM

EEPROM Registers.

Table 23. EEPROM Register Index

Offset (Hex)	Register Module Base Address: 0x600	
	Register Name	**Register Description**
0x0	EEPROM_CNFG	EEPROM Configuration
0x8	EEPROM_ADDR_CNFG	EEPROM Address Configuration
0x9	EEPROM_EVENT	EEPROM Events
0xA	EEPROM_ERR_CNT_EVENT	EEPROM Error Counter Status

3.19.1 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.
19:13	reserved	RO	0x0	Reserved
12	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.

EEPROM_CNFG Bit Field Descriptions				
Bit Field	**Field Name**	**Field Type**	**Default Value**	**Description**
11	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
10:7	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
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.

3.19.2 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)

3.19.3 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.
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.

3.19.4 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. Revision History

Revision	Date	Description
0.03	Apr 26, 2024	▪ Updated the front matter. ▪ Completed numerous changes to Registers. ▪ Completed other minor changes.
0.02	Mar 3, 2023	▪ Added Serial Interfaces section. ▪ Updated the headings in Table 5.
0.01	Feb 24, 2023	This is the first preliminary release of the programming guide.