

RC22312, RC22308

FemtoClock<sup>®</sup>3 Multi-Frequency Clock Synthesizer

The RC22312/RC22308 is an ultra-low phase noise multi-frequency clock synthesizer and digitally controlled oscillator (DCO). This flexible, low-power device outputs clocks with 50fs RMS jitter supporting SerDes operating at 112Gbps.

This document provides programming information for software engineers to use the RC22312/RC22308.

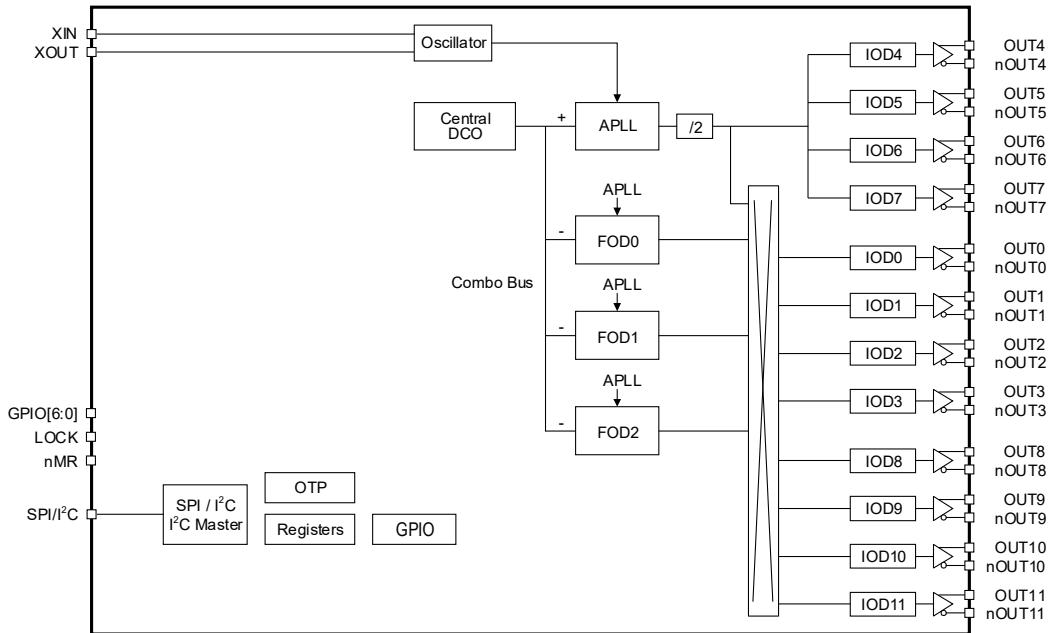


Figure 1. RC22312 Block Diagram

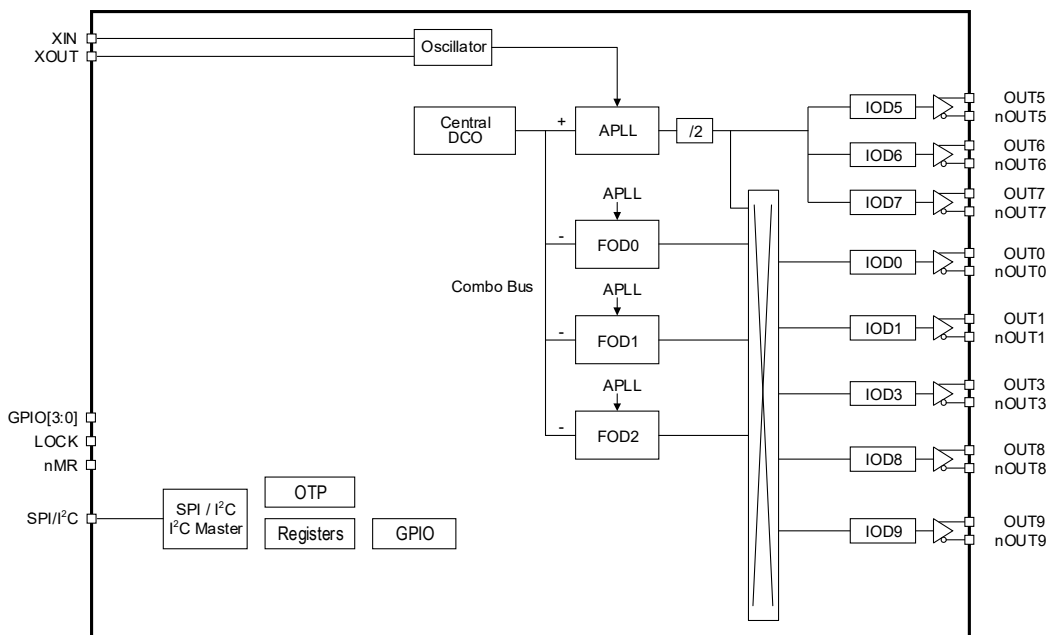


Figure 2. RC22308 Block Diagram

Note: Device specifications and ordering information are available in the RC22312/RC22308 Datasheet.

## Contents

|  |           |
|--|-----------|
| <b>1. Functional Description</b>                   | <b>7</b>  |
| 1.1 Overview                                       | 7         |
| 1.2 Device Frequency Reference                     | 7         |
| 1.3 Analog PLL                                     | 8         |
| 1.4 Fractional Output Dividers                     | 8         |
| 1.4.1 FOD Synthesizer Mode                         | 8         |
| 1.4.2 FOD Digitally Controlled Oscillator Mode     | 8         |
| 1.4.3 FOD Synchronous Mode                         | 9         |
| 1.5 Divider Synchronization                        | 9         |
| 1.6 Crystal Monitor                                | 9         |
| 1.7 Central DCO                                    | 10        |
| 1.7.1 Central DCO State Machine                    | 10        |
| 1.7.2 Freerun State                                | 10        |
| 1.7.3 Write Frequency State                        | 10        |
| 1.8 Clock Output Paths                             | 10        |
| 1.8.1 Clock Output Enable                          | 10        |
| 1.8.2 Integer Output Dividers                      | 12        |
| 1.8.3 Clock Output Path for OUT[7:4]               | 12        |
| 1.8.4 Output Clock Path for OUT[3:0] and OUT[11:8] | 13        |
| 1.8.5 Clock Output Buffers                         | 13        |
| 1.9 Status and Control                             | 13        |
| 1.10 Power-On Reset and Reset Controller           | 13        |
| 1.10.1 Master Reset Sequence                       | 14        |
| 1.10.2 Soft Reset Sequence                         | 15        |
| <b>2. Serial Interfaces</b>                        | <b>16</b> |
| 2.1 Paging   | 16        |
| 2.2 I2C Slave                                      | 16        |
| 2.2.1 I2C 1-byte (1B) Addressing Example           | 17        |
| 2.2.2 I2C 2-byte (2B) Addressing Example           | 18        |
| 2.3 SPI Slave                                      | 18        |
| 2.3.1 SPI 1-byte (1B) Addressing Example           | 19        |
| 2.3.2 SPI 2-byte (2B) Addressing Example           | 19        |
| <b>3. Register Set Descriptions</b>                | <b>20</b> |
| 3.1 GLOBAL   | 21        |
| 3.1.1 VENDOR_ID                                    | 21        |
| 3.1.2 DEVICE_ID                                    | 21        |
| 3.1.3 DEVICE_REV                                   | 22        |
| 3.1.4 DEVICE_PGM                                   | 22        |
| 3.1.5 DEVICE_CNFG                                  | 22        |
| 3.1.6 MISC_CNFG                                    | 23        |
| 3.1.7 SCRATCH_CNFG                                 | 24        |
| 3.1.8 MISC_CTRL                                    | 24        |
| 3.1.9 STARTUP_STS                                  | 24        |
| 3.1.10 DEVICE_STS                                  | 25        |
| 3.2 INT  | 25        |
| 3.2.1 INT_EN_CTRL                                  | 26        |
| 3.2.2 INT_STS                                      | 26        |
| 3.3 SSI  | 27        |
| 3.3.1 I2C_FLTR_CNFG                                | 27        |
| 3.3.2 I2C_TIMING_CNFG                              | 28        |
| 3.3.3 I2C_ADDR_CNFG                                | 28        |

|        |                             |    |
|--------|-----------------------------|----|
| 3.3.4  | SPI_CNFG                    | 28 |
| 3.3.5  | SSI_GLOBAL_CNFG             | 28 |
| 3.3.6  | SCL_SCLK_PAD_CNFG           | 29 |
| 3.3.7  | SDA_SDIO_PAD_CNFG           | 29 |
| 3.3.8  | SDO_A1_PAD_CNFG             | 30 |
| 3.3.9  | NCS_A0_PAD_CNFG             | 30 |
| 3.3.10 | I2C_EVENT                   | 30 |
| 3.3.11 | I2C_CRC_ERR_CNT_EVENT       | 30 |
| 3.4    | XO                          | 31 |
| 3.4.1  | XO_CNFG                     | 31 |
| 3.5    | SYSDIV                      | 32 |
| 3.5.1  | SYS_DIV_INT_CNFG            | 32 |
| 3.5.2  | COUNTER_1US_CNFG            | 32 |
| 3.5.3  | SYS_DIV_EN_CTRL             | 32 |
| 3.6    | GPIO                        | 33 |
| 3.6.1  | GPIO_CNFG                   | 33 |
| 3.6.2  | GPIO_DEGLITCH_CNFG          | 36 |
| 3.6.3  | GPIO_PAD_CNFG               | 36 |
| 3.6.4  | GPIO_STS                    | 37 |
| 3.7    | OUTBUF                      | 37 |
| 3.7.1  | OUT_CNFG                    | 37 |
| 3.7.2  | OUT_MODE_CNFG               | 37 |
| 3.7.3  | OUT_HCSL_CMOS_CNFG          | 38 |
| 3.7.4  | OUT_LVDS_CNFG               | 39 |
| 3.7.5  | OUT_EN_CTRL                 | 39 |
| 3.7.6  | OUT_SPARE                   | 40 |
| 3.8    | TDCAPLL                     | 40 |
| 3.8.1  | TDC_FB_DIV_INT_CNFG         | 40 |
| 3.8.2  | TDC_REF_DIV_CNFG            | 40 |
| 3.8.3  | TDC_ENABLE_CTRL             | 41 |
| 3.9    | XTALMON                     | 41 |
| 3.9.1  | XTALMON_NOMINAL_MARGIN_CNFG | 41 |
| 3.9.2  | XTALMON_WINDOW_CNFG         | 42 |
| 3.9.3  | XTALMON_QUAL_CNFG           | 42 |
| 3.9.4  | XTALMON_CTRL                | 42 |
| 3.9.5  | XTALMON_EVENT               | 43 |
| 3.9.6  | XTALMON_CNT_EVENT           | 43 |
| 3.9.7  | XTALMON_STS                 | 43 |
| 3.10   | APLL                        | 43 |
| 3.10.1 | LPF_TC_CNFG                 | 44 |
| 3.10.2 | APLL_FB_DIV_FRAC_CNFG       | 44 |
| 3.10.3 | APLL_FB_DIV_INT_CNFG        | 44 |
| 3.10.4 | APLL_DCD_CAL_CNFG           | 44 |
| 3.11   | IOD                         | 45 |
| 3.11.1 | IOD_DIV_CNFG                | 45 |
| 3.11.2 | IOD_PHASE_CNFG              | 45 |
| 3.11.3 | IOD_CNFG                    | 46 |
| 3.11.4 | IOD_EN_CTRL                 | 46 |
| 3.11.5 | IOD_PHASE_EN_CTRL           | 46 |
| 3.12   | FOD                         | 47 |
| 3.12.1 | FOD_CNFG                    | 47 |
| 3.12.2 | FOD_PHASE_CNFG              | 47 |
| 3.12.3 | FOD_NUM_CNFG                | 48 |

|           |                               |           |
|-----------|-------------------------------|-----------|
| 3.12.4    | FOD_DEN_CNFG .....            | 48        |
| 3.12.5    | FOD_DIV_CNFG .....            | 48        |
| 3.12.6    | FOD_EN_CTRL .....             | 48        |
| 3.12.7    | FOD_PHASE_EN_CTRL .....       | 49        |
| 3.12.8    | FOD_WRITE_FREQ_CTRL .....     | 49        |
| 3.13      | LDO .....                     | 49        |
| 3.13.1    | FOD_LDO_CNFG .....            | 49        |
| 3.14      | DPLL .....                    | 49        |
| 3.14.1    | DPLL_MODE_CNFG .....          | 50        |
| 3.14.2    | DPLL_XTAL_OFFSET_CNFG .....   | 50        |
| 3.14.3    | DPLL_WR_FREQ_CTRL .....       | 50        |
| 3.15      | EEPROM .....                  | 50        |
| 3.15.1    | EEPROM_CNFG .....             | 51        |
| 3.15.2    | EEPROM_ADDR_CNFG .....        | 52        |
| 3.15.3    | EEPROM_EVENT .....            | 52        |
| 3.15.4    | EEPROM_ERR_CNT_EVENT .....    | 53        |
| <b>4.</b> | <b>Revision History .....</b> | <b>54</b> |

## Figures

|   |    |
|---|----|
| Figure 1. RC22312 Block Diagram .....                       | 1  |
| Figure 2. RC22308 Block Diagram .....                       | 1  |
| Figure 3. RC22312 Functional Block Diagram .....            | 7  |
| Figure 4. Crystal Monitor and TDC APLL .....                | 9  |
| Figure 5. Synthesizer or DCO State Machine .....            | 10 |
| Figure 6. Clock Path for OUT[7:4] .....                     | 12 |
| Figure 7. Clock Path for OUT[3:0] and OUT[11:8] .....       | 13 |
| Figure 8. Master Reset Sequence Initiation .....            | 14 |
| Figure 9. Register Addressing Modes Using Serial Port ..... | 16 |
| Figure 10. I2C Slave Sequencing .....                       | 17 |
| Figure 11. SPI Sequencing .....                             | 18 |

## Tables

|   |    |
|---|----|
| Table 1. DCO States .....   | 10 |
| Table 2. Clock Output Enable Control Truth Table .....                    | 11 |
| Table 3. GPIO Clock Output Enable Control Hierarchy and Truth Table ..... | 11 |
| Table 4. Register Set Module Index .....                                  | 20 |
| Table 5. GLOBAL Register Index .....                                      | 21 |
| Table 6. INT Register Index .....   | 25 |
| Table 7. SSI Register Index .....   | 27 |
| Table 8. XO Register Index .....  | 31 |
| Table 9. SYSDIV Register Index .....                                      | 32 |
| Table 10. GPIO Register Index .....                                       | 33 |
| Table 11. OUTBUF Register Index .....                                     | 37 |
| Table 12. TDCAPLL Register Index .....                                    | 40 |
| Table 13. XTALMON Register Index .....                                    | 41 |
| Table 14. APLL Register Index .....                                       | 43 |
| Table 15. IOD Register Index .....  | 45 |
| Table 16. FOD Register Index .....  | 47 |
| Table 17. LDO Register Index .....  | 49 |
| Table 18. DPLL Register Index .....                                       | 49 |
| Table 19. EEPROM Register Index .....                                     | 50 |

# 1. Functional Description

## 1.1 Overview

The RC22312/22308 is an ultra-low phase noise multi-frequency synthesizer and digitally controlled oscillator (DCO). This flexible, low-power device outputs clocks with 25fs RMS jitter supporting 112Gbps and 224Gbps SerDes.

The RC22312 has 12 differential clock outputs (OUTx). The RC22308 has eight differential clock outputs. Both devices provide a DCO, 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 RC22312, and see Figure 2 for a block diagram of the RC22308.

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 RC22312, this diagram is also representative of the architecture of RC22308.

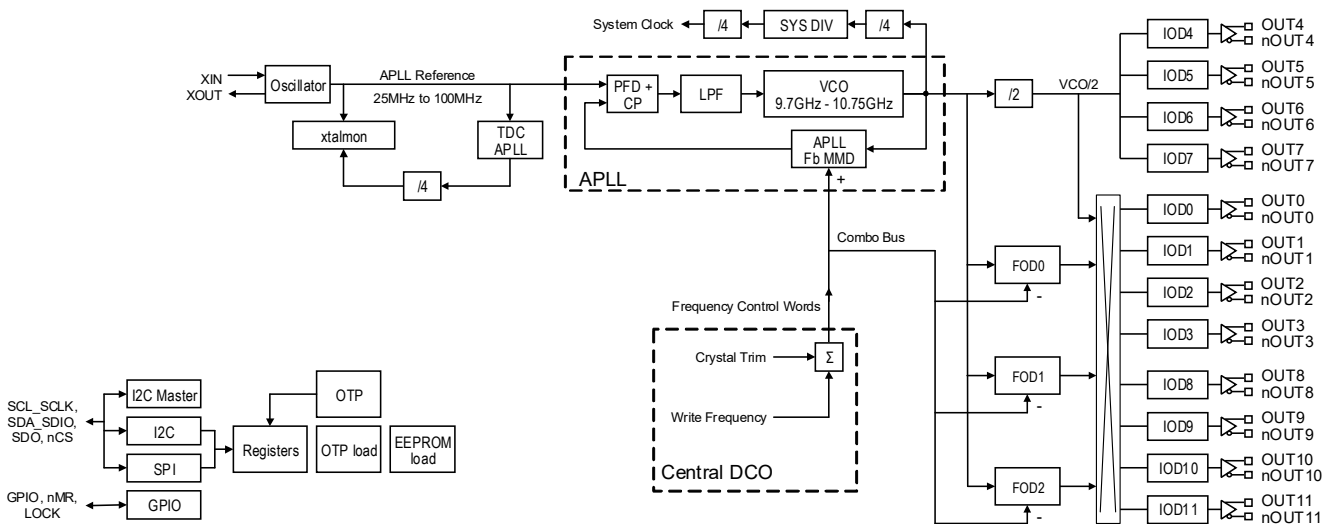


Figure 3. RC22312 Functional Block Diagram

## 1.2 Device Frequency Reference

The RC22312/22308 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 RC22312/22308 *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 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 that a filtering algorithm can support.

## 1.3 Analog PLL

The APLL is configured using register fields in the APLL section of the [Register Set Descriptions](#).

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 Central DCO can steer the FFO of the APLL using the frequency control words (FCW) it outputs on the Combo Bus as shown in the equation below, see [Figure 3](#). When the DCO is in the [Write Frequency State](#), the FCW is controlled via the [write\\_freq](#) register field.

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

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 register fields in the FOD section of the [Register Set Descriptions](#).

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: [FOD Synthesizer Mode](#), [FOD Digitally Controlled Oscillator Mode](#), and [FOD Synchronous Mode](#). All FOD modes divide the VCO clock, the modes differ in their intended applications, how the divide ratios are specified.

Unused FODs can be powered down to reduce power consumption by setting the respective [fod\\_enable](#) and [fod\\_en\\_ldo](#) register bits to 0x0.

### 1.4.1 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 Central DCO via the [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_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.

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

An FOD in synchronous mode will faithfully follow frequency steering applied to the VCO by the DPLL, or by external software via the `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 and FODs 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](#)).

## 1.6 Crystal Monitor

The crystal monitor is configured using register fields in the XTALMON section of the [Register Set Descriptions](#).

The crystal monitor (XTALMON) is a LOS monitor that monitors for loss of signal (LOS). The timing reference for the crystal monitor is the measuring clock that is traceable to the device frequency reference via the TDC APLL as shown in [Figure 4](#).

The TDC APLL synthesizer is a ring oscillator with frequency range 700MHz to 1.2GHz. In the event that the clock input to the TDC APLL is not toggling, the ring oscillator will generate a 700Mz clock that enables the XTALMON to detect the loss of the device frequency reference signal.

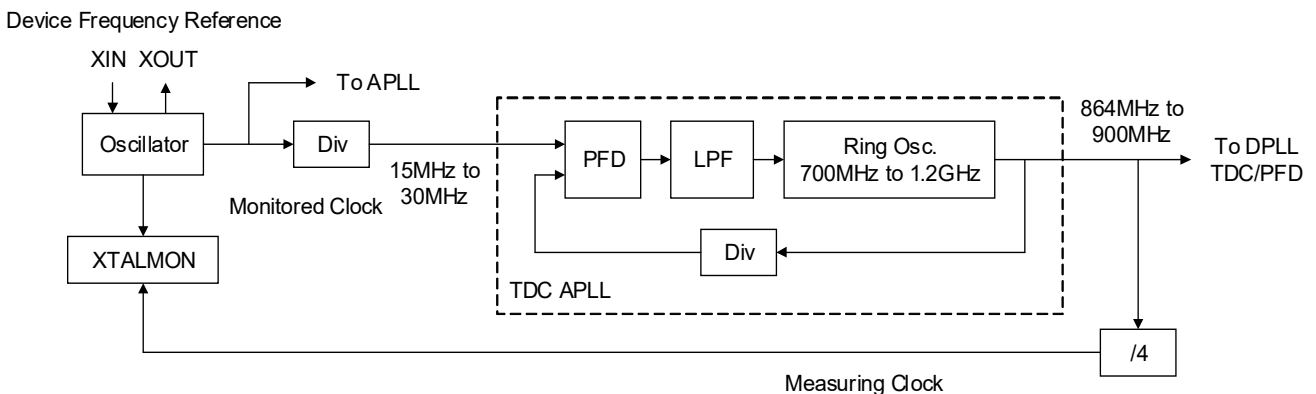


Figure 4. Crystal Monitor and TDC APLL

## 1.7 Central DCO

The Central DCO is configured using register fields in the DPLL section of the [Register Set Descriptions](#).

External software can use the APLL DCO to steer the frequency of the APLL using the [write\\_freq](#) register field as shown in [Figure 3](#).

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 by compensating for a known FFO of the frequency reference.

### 1.7.1 Central DCO State Machine

Table 1. DCO States

| DCO State       | dpll_mode               |
|-----------------|-------------------------|
| Freerun         | 0x0, 0x6                |
| Write Frequency | 0x3                     |
| Reserved        | 0x1, 0x2, 0x4, 0x5, 0x7 |

### 1.7.2 Freerun State

In the Freerun state, the [write\\_freq](#) term output by the Central DCO is 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 DCO will be in the Freerun state (see [Figure 5](#)). There are two settings of the [dpll\\_mode](#) register field that will force the DPLL into the Freerun state (see [Table 1](#)).

### 1.7.3 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 DCO can be forced into the Write Frequency state by the following setting: [dpll\\_mode](#) = 0x3 (see [Table 1](#)).

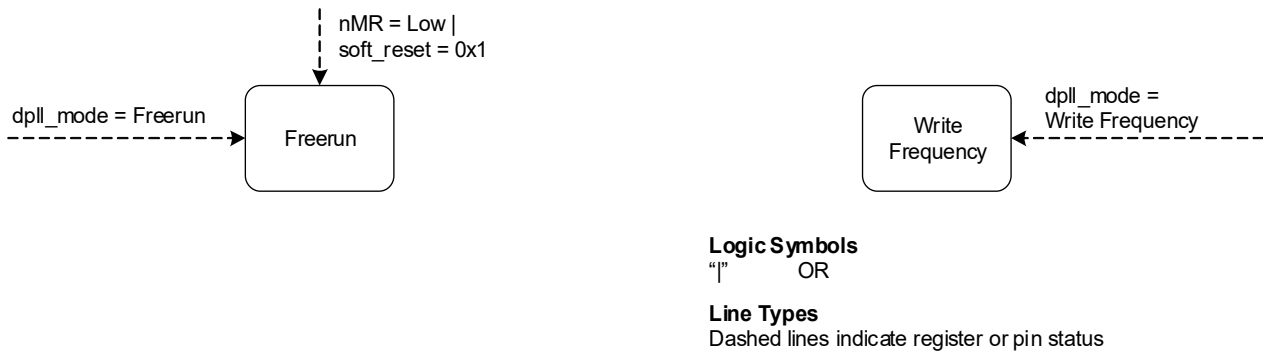


Figure 5. Synthesizer or DCO State Machine

## 1.8 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.

### 1.8.1 Clock Output Enable

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

When [global\\_oe](#) = 0x0, all clock outputs are disabled. When [global\\_oe](#) = 0x1, clock outputs with [oe\\_source\\_sel](#) = 0x1 can be enabled or disabled individually using the respective [out\\_driver\\_en](#) register bit. When [global\\_oe](#) = 0x1,

clock outputs with `oe_source_sel = 0x0`, can be enabled or disabled in groups or individually under GPIO control. Please see [Table 2](#) for a summary.

**Table 2. Clock Output Enable Control Truth Table**

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

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

GPIOs can be assigned one of the following clock output enable functions using the `gpio_func` register field: Global OE, Group OE, or OE. When a GPIO is assigned the Global OE function (`gpio_func = 0x4C`), that pin controls the output enable for all clock outputs with `oe_source_sel = 0x0`. When a GPIO is assigned the Group OE function (`gpio_func = 0x4D to 0x4F`) and no GPIO is assigned the Global OE function, that pin controls the output enable for all clock outputs in the group with `oe_source_sel = 0x0`. When a GPIO is assigned the OE function (`gpio_func = 0x50 to 0x5B`) and no GPIO is assigned the Global OE function or the Group OE function, that pin controls the output enable for its assigned clock output if the clock output has `oe_source_sel = 0x0`. Please see [Table 3](#) for a summary.

**Table 3. GPIO Clock Output Enable Control Hierarchy and Truth Table [1]**

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

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

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

When a GPIO is assigned an output enable function it should be configured with `gpio_resync = 0x0` and `gpio_deglitch_bypass = 0x1`; this disables GPIO resynchronization and bypasses the GPIO deglitcher ensuring

lowest latency. When so configured, the maximum time delay from the time the voltage level on a GPIO input changes state until the clock output is enabled or disabled is 14ns plus 4 periods of the output clock.

### 1.8.2 Integer Output Dividers

The IODs are configured using register fields in the IOD section of the [Register Set Descriptions](#) 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.

#### 1.8.2.1 IOD Phase Adjustment

The phase of each IOD output clock can be independently adjusted using the signed 16-bit [iod\\_phase\\_config](#) register field. IOD phase adjustments are specified in periods of the IOD input clock.

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

IOD phase adjustments are triggered by writing the [iod\\_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).

### 1.8.3 Clock Output Path for OUT[7:4]

The clock source selection for OUT[7:4] is illustrated in [Figure 6](#).

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.

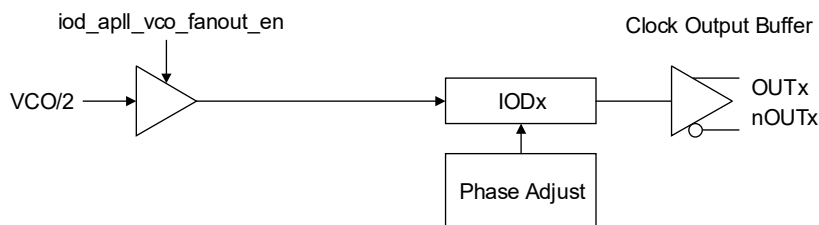


Figure 6. Clock Path for OUT[7:4]

### 1.8.4 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 7.

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`.

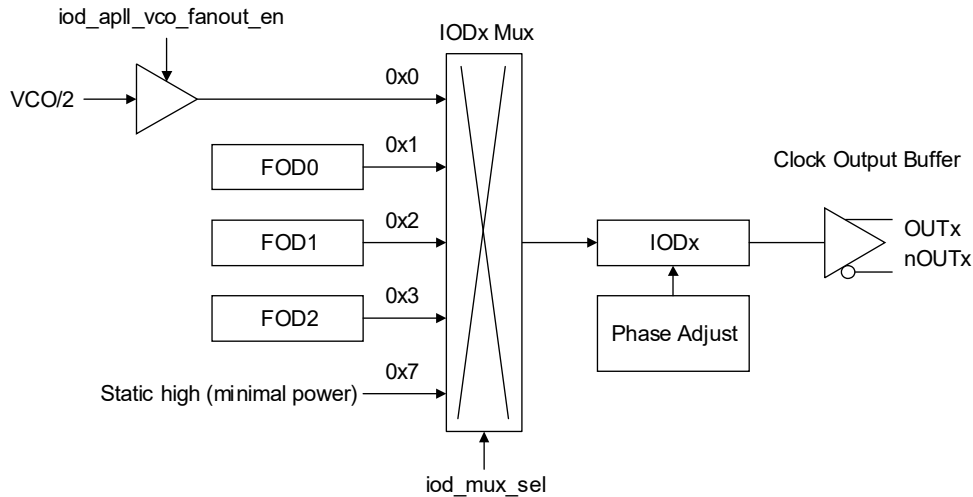


Figure 7. Clock Path for OUT[3:0] and OUT[11:8]

### 1.8.5 Clock Output Buffers

The clock output buffers are configured using register fields in the OUTBUF section of the Register Set Descriptions.

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%.

## 1.9 Status and Control

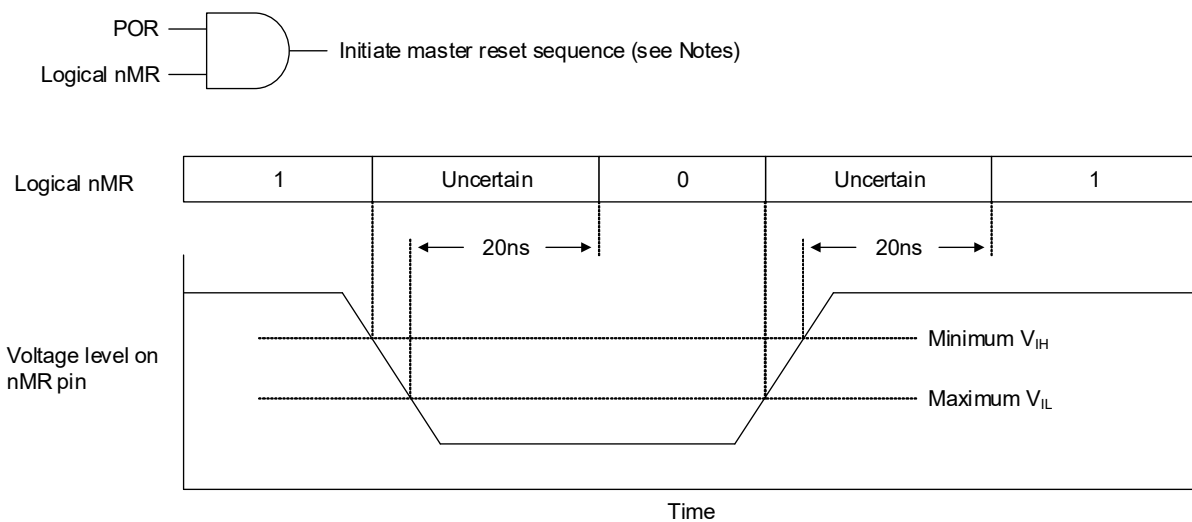
All control and status registers (CSR) are accessed through a 1MHz I<sup>2</sup>C or 20MHz SPI slave microprocessor interface. The device can automatically load a configuration from internal one time programmable (OTP) memory. Alternatively, the I<sup>2</sup>C master interface can automatically load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

### 1.10 Power-On Reset and Reset Controller

There are no power supply sequencing requirements; however, if  $V_{DD0x}$  or  $V_{DD\_CLK}$  reach 90% of  $V_{DD}$  nominal after the later of  $V_{DD\_VCO}$  or  $V_{DD33\_DIA}$  then a soft reset or a master reset must be initiated to ensure the 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 8). 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 8. 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.

### 1.10.1 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<sup>2</sup>C 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.
  - a. Start the Central DCO state machine (see [Central DCO State Machine](#) and [Table 1](#)).
7. Set `rst_done_sts` to 0x1.

### 1.10.2 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<sup>2</sup>C 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<sup>2</sup>C 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. 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 Central DCO state machine (see [Central DCO State Machine](#) and [Table 1](#)).
  - 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

I<sup>2</sup>C or SPI operation is selected by the `ssi_enable` register field, which defaults to I<sup>2</sup>C 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 9 shows how page register and offset bytes from each serial transaction interact to address a register within the RC22312, RC22308.

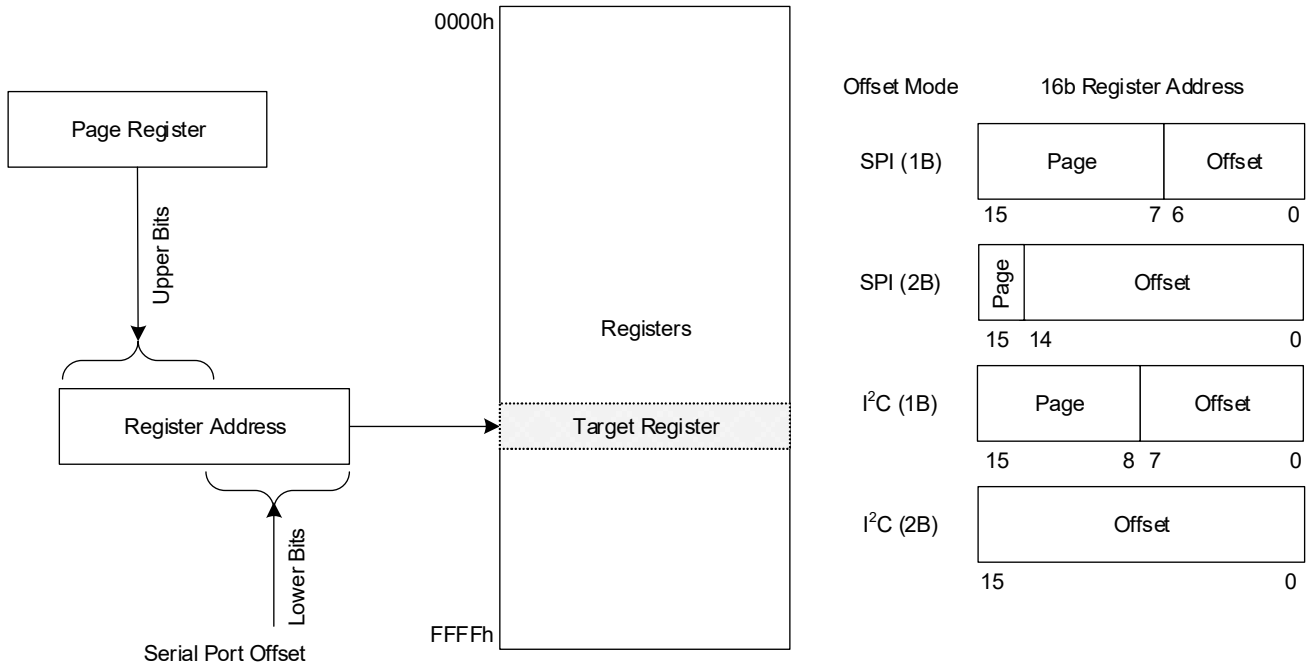


Figure 9. Register Addressing Modes Using Serial Port

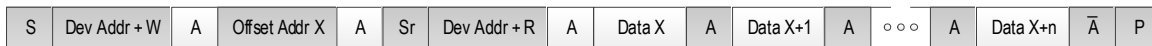
### 2.2 I<sup>2</sup>C Slave

The I<sup>2</sup>C slave protocol of the RC22312, RC22308 complies with the I<sup>2</sup>C 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 10 shows the sequence of states on the I<sup>2</sup>C SDA signal for the supported modes of operation.

The Dev Addr shown in the figure represents the I<sup>2</sup>C bus address of the device; this 7-bit value in the `i2c_addr` register field defaults to 0x09. To use a different I<sup>2</sup>C bus address, an OTP configuration must alter the value of one or both of the `i2c_addr` and `i2c_addr_sel` register fields. The value in the `i2c_addr_sel` register field determines if the I<sup>2</sup>C bus address can be controlled through the nCS\_A0, SDO\_A1 or GPIO pins.

Sequential 1-byte Read



Sequential 1-byte Write



Sequential 2-byte Read



Sequential 2-byte Write



- From master to slave
- From slave to master
- S = Start
- Sr = Repeated start
- A = Acknowledge
- $\bar{A}$  = Non-acknowledge
- P = Stop

Figure 10. I<sup>2</sup>C Slave Sequencing

The selection of 1-byte (1B) or 2-byte (2B) offset addressing must also be configured using the `ssi_addr_size` register field. These offsets are used in conjunction with the page register to access registers internal to the device (see Figure 9). Because the I<sup>2</sup>C 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:* I<sup>2</sup>C 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 I<sup>2</sup>C 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.

I<sup>2</sup>C interface timing is shown in the RC22312, RC22308 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 I<sup>2</sup>C interface operating at 1MHz supports a DCO update rate of approximately 16k updates per second.

### 2.2.1 I<sup>2</sup>C 1-byte (1B) Addressing Example

RC22312, RC22308 I<sup>2</sup>C 7-bit I<sup>2</sup>C 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 I<sup>2</sup>C 2-byte (2B) Addressing Example

RC22312, RC22308 I<sup>2</sup>C 7-bit I<sup>2</sup>C 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 RC22312, RC22308 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 RC22312, RC22308) and data out signals (SDI and SDO respectively). In 3-wire mode, the SDIO signal is used as a single, bidirectional data signal.



Figure 11. SPI Sequencing

Figure 11 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 9)
- 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 *RC22312, RC22308 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. Register Set Descriptions

Table 4. Register Set Module Index

| Module Base Address (Hex) | Name       | Module Description  | Link                    |
|---------------------------|------------|---|-------------------------|
| 0x0                       | GLOBAL     | Global Control and Status Registers   | <a href="#">GLOBAL</a>  |
| 0x40                      | INT        | Interrupt Registers   | <a href="#">INT</a>     |
| 0x50                      | SSI        | Slave Serial Interface Registers  | <a href="#">SSI</a>     |
| 0x60                      | XO         | Crystal Oscillator, Input Buffer and Reference Select Registers                                     | <a href="#">XO</a>      |
| 0x94                      | SYSDIV     | System Clock Divider Registers  | <a href="#">SYSDIV</a>  |
| 0xA0                      | GPIO[0]    | General Purpose IO Registers<br>Instances 0-7 apply to GPIO0 to GPIO9<br>Instance 8 applies to LOCK | <a href="#">GPIO</a>    |
| 0xA8                      | GPIO[1]    | Same as GPIO[0]   | <a href="#">GPIO</a>    |
| 0xB0                      | GPIO[2]    | Same as GPIO[0]   | <a href="#">GPIO</a>    |
| 0xB8                      | GPIO[3]    | Same as GPIO[0]   | <a href="#">GPIO</a>    |
| 0xC0                      | GPIO[4]    | Same as GPIO[0]   | <a href="#">GPIO</a>    |
| 0xC8                      | GPIO[5]    | Same as GPIO[0]   | <a href="#">GPIO</a>    |
| 0xD0                      | GPIO[6]    | Same as GPIO[0]   | <a href="#">GPIO</a>    |
| 0xD8                      | GPIO[7]    | Same as GPIO[0]   | <a href="#">GPIO</a>    |
| 0xE0                      | GPIO[8]    | Same as GPIO[0]   | <a href="#">GPIO</a>    |
| 0x100                     | OUTBUF[0]  | Output Buffer Registers   | <a href="#">OUTBUF</a>  |
| 0x108                     | OUTBUF[1]  | Same as OUTBUF[0]   | <a href="#">OUTBUF</a>  |
| 0x110                     | OUTBUF[2]  | Same as OUTBUF[0]   | <a href="#">OUTBUF</a>  |
| 0x118                     | OUTBUF[3]  | Same as OUTBUF[0]   | <a href="#">OUTBUF</a>  |
| 0x120                     | OUTBUF[4]  | Same as OUTBUF[0]   | <a href="#">OUTBUF</a>  |
| 0x128                     | OUTBUF[5]  | Same as OUTBUF[0]   | <a href="#">OUTBUF</a>  |
| 0x130                     | OUTBUF[6]  | Same as OUTBUF[0]   | <a href="#">OUTBUF</a>  |
| 0x138                     | OUTBUF[7]  | Same as OUTBUF[0]   | <a href="#">OUTBUF</a>  |
| 0x140                     | OUTBUF[8]  | Same as OUTBUF[0]   | <a href="#">OUTBUF</a>  |
| 0x148                     | OUTBUF[9]  | Same as OUTBUF[0]   | <a href="#">OUTBUF</a>  |
| 0x150                     | OUTBUF[10] | Same as OUTBUF[0]   | <a href="#">OUTBUF</a>  |
| 0x158                     | OUTBUF[11] | Same as OUTBUF[0]   | <a href="#">OUTBUF</a>  |
| 0x160                     | TDCAPLL    | TDC APLL Registers  | <a href="#">TDCAPLL</a> |
| 0x170                     | XTALMON    | XTAL Monitor Registers (xtalmon[0] = XIN, xtalmon[1]=REF4)  | <a href="#">XTALMON</a> |
| 0x240                     | APLL       | APLL Registers  | <a href="#">APLL</a>    |
| 0x2A0                     | IOD[0]     | Integer Output Divider Registers  | <a href="#">IOD</a>     |
| 0x2B0                     | IOD[1]     | Same as IOD[0]  | <a href="#">IOD</a>     |
| 0x2C0                     | IOD[2]     | Same as IOD[0]  | <a href="#">IOD</a>     |
| 0x2D0                     | IOD[3]     | Same as IOD[0]  | <a href="#">IOD</a>     |
| 0x2E0                     | IOD[4]     | Same as IOD[0]  | <a href="#">IOD</a>     |
| 0x2F0                     | IOD[5]     | Same as IOD[0]  | <a href="#">IOD</a>     |
| 0x300                     | IOD[6]     | Same as IOD[0]  | <a href="#">IOD</a>     |
| 0x310                     | IOD[7]     | Same as IOD[0]  | <a href="#">IOD</a>     |
| 0x320                     | IOD[8]     | Same as IOD[0]  | <a href="#">IOD</a>     |
| 0x330                     | IOD[9]     | Same as IOD[0]  | <a href="#">IOD</a>     |
| 0x340                     | IOD[10]    | Same as IOD[0]  | <a href="#">IOD</a>     |

Table 4. Register Set Module Index

| Module Base Address (Hex) | Name    | Module Description                  | Link                   |
|---------------------------|---------|-------------------------------------|------------------------|
| 0x350                     | IOD[11] | Same as IOD[0]                      | <a href="#">IOD</a>    |
| 0x400                     | FOD[0]  | Fractional Output Divider Registers | <a href="#">FOD</a>    |
| 0x440                     | FOD[1]  | Same as FOD[0]                      | <a href="#">FOD</a>    |
| 0x480                     | FOD[2]  | Same as FOD[0]                      | <a href="#">FOD</a>    |
| 0x4C0                     | LDO     | LDO Registers                       | <a href="#">LDO</a>    |
| 0x500                     | DPLL    | DPLL Registers                      | <a href="#">DPLL</a>   |
| 0x600                     | EEPROM  | EEPROM Registers                    | <a href="#">EEPROM</a> |

### 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          | <a href="#">VENDOR_ID</a>         | Vendor ID and Device Type    |
| 0x2          | <a href="#">DEVICE_ID</a>         | Device ID                    |
| 0x4          | <a href="#">DEVICE_REV</a>        | Device Revisions and Font ID |
| 0x6          | <a href="#">DEVICE_PGM</a>        | Device Dash Code             |
| 0x8          | <a href="#">DEVICE_CNFG</a>       | Device Configuration         |
| 0xC          | <a href="#">MISC_CNFG</a>         | Reset Configuration          |
| 0x10         | <a href="#">SCRATCH_CNFG</a>      | Scratch register             |
| 0x14         | <a href="#">MISC_CTRL</a>         | Reset Control                |
| 0x20         | <a href="#">STARTUP_STS</a>       | Startup status               |
| 0x24         | <a href="#">DEVICE_STS</a>        | 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 writeable 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:<br>0x0 = Font 0<br>0x2 = Font 1<br>0x3 = Font 2<br>0x4 = Font 3<br>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.<br>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.<br>0x0 = i2c_addr[2], i2c_addr[1], i2c_addr[0]<br>0x1 = i2c_addr[2], i2c_addr[1], SDO<br>0x2 = i2c_addr[2], i2c_addr[1], nCS<br>0x3 = i2c_addr[2], SDO, nCS<br>0x4 = i2c_addr[2], gpio[1], gpio[0]<br>0x5 = gpio[2], i2c_addr[1], gpio[0]<br>0x6 = gpio[2], i2c_addr[1], i2c_addr[0]<br>0x7 = gpio[2], SDO, nCS<br>0x8 = gpio[2], gpio[1], SDO<br>0x9 = gpio[2], gpio[1], nCS |
|                                    | i2c_addr_sel (continued) |            |               | 0xA = i2c_addr[2], SDO, nCS<br>0xB = i2c_addr[2], SDO, gpio[0]<br>0xC = i2c_addr[2], gpio[1], gpio[0]<br>0xD = SDO, gpio[1], gpio[0]<br>0xE = nCS, gpio[1], gpio[0]<br>0xF = gpio[2], gpio[1], gpio[0]  |

| DEVICE_CNFG Bit Field Descriptions |            |            |               |  |
|------------------------------------|------------|------------|---------------|--|
| Bit Field                          | Field Name | Field Type | Default Value | Description  |
| 3:0                                | config_sel | RW         | 0x0           | OTP Config Select. Controls source of OTP config selection from a combination of pins at startup. gpio[0], and gpio[1] refers to any GPIO pin which has gpio_startup_mode set to 0x0, and 0x1 respectively.<br>0x0 = User config 0<br>0x1 = User config 1<br>0x2 = User config 2<br>0x3 = User config 3<br>0x4 = SDA_SDIO, SCL_SCLK<br>0x5 = SDA_SDIO, SDO_A1<br>0x6 = SDA_SDIO, nCS_A0<br>0x7 = SDA_SDIO, gpio[0]<br>0x8 = SCL_SCLK, SDO_A1<br>0x9 = SCL_SCLK, nCS_A0 |
| config_sel (continued)             |            |            |               | 0xA = SCL_SCLK, gpio[0]<br>0xB = SDO_A1, gpio[0]<br>0xC = nCS_A0, SDO_A1<br>0xD = nCS_A0, gpio[0]<br>0xE = gpio[1], gpio[0]<br>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<br>0x0 = 1 ms<br>0x1 = 250 us<br>0x2 = 500 us<br>0x3 = 2.5 ms<br>0x4 = 5 ms<br>0x5 = 10 ms<br>0x6 = 20 ms<br>0x7 = Reserved   |
| 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.<br>0x0 = Clock outputs are disabled until APLL lock asserts (or times out)<br>0x1 = Clock outputs are disabled until APLL lock asserts (or times out) and DCD and FOD calibration is done<br>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.<br>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<br>0x0 = Don't relatch inputs on soft reset<br>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.<br>0x0 = Soft reset recalibrates the VCO, relocks the APLL, recalibrates the DCDs, and resyns all dividers<br>0x1 = Resyns 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.<br>0x0 = All outputs are disabled<br>0x1 = Output enables are controlled by their individual settings           |
| 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.<br>0x0 = Low<br>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.<br>0x0 = Low<br>0x1 = High |

| STARTUP_STS Bit Field Descriptions |                       |            |               |  |
|------------------------------------|-----------------------|------------|---------------|--|
| Bit Field                          | Field Name            | Field Type | Default Value | Description  |
| 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.<br>0x0 = Low<br>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.<br>0x0 = Low<br>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.<br>0x0 = Low<br>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.<br>0x0 = Low<br>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                                 | eeeprom_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          | <a href="#">INT_EN_CTRL</a>        | Interrupt Enable Configuration |
| 0x8          | <a href="#">INT_STS</a>            | 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                             |
| 37:36                              | reserved              | RO         | 0x0           | Reserved   |
| 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            |
| 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      |
| 30:25                              | reserved              | RW         | 0x0           | Reserved   |
| 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 |
| 23:15                              | reserved              | RW         | 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                  |
| 13:2                               | 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.  |
| 37:36                          | reserved               | RO         | 0x0           | Reserved  |
| 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   |
| 30:25                          | reserved             | RO         | 0x0           | Reserved  |
| 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. |
| 23:15                          | 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.  |
| 13:2                           | 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                    |
| 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_ftr | 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.<br>0x0 = SDO is driven on opposite SCLK edge than the sampling edge<br>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.<br>0x0 = SDI is sampled on rising SCLK edge<br>0x1 = SDI is sampled on falling SCLK edge                   |
| 0                               | spi_3wire   | RW         | 0x0           | Select SPI 3 or 4-wire mode.<br>0x0 = Normal 4-wire SPI. Data is received on SDA_SDIO, and transmitted on the SDO.<br>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.<br>0x0 = CRC disabled<br>0x1 = CRC enabled |

| SSI_GLOBAL_CNFG Bit Field Descriptions |               |            |               |  |
|--|---------------|------------|---------------|--|
| Bit Field                              | Field Name    | Field Type | Default Value | Description  |
| 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.<br>0x0 = 1-byte address<br>0x1 = 2-byte address |
| 1:0                                    | ssi_enable    | RW         | 0x1           | SSI mode. Selects the serial interface mode.<br>0x0 = SSI is disabled<br>0x1 = SSI is in I2C mode<br>0x2 = SSI is in SPI mode<br>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<br>0x0 = Open drain Output mode. Standard mode.<br>0x1 = Open drain Output mode. Fast mode.<br>0x2 = Reserved<br>0x3 = Open drain Output mode. Fast mode plus.<br>0x4 = CMOS Output mode and power supply of 3.3V.<br>0x5 = Reserved<br>0x6 = CMOS Output mode and power supply of 1.8V.<br>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<br>0x0 = Open drain Output mode. Standard mode.<br>0x1 = Open drain Output mode. Fast mode.<br>0x2 = Reserved<br>0x3 = Open drain Output mode. Fast mode plus.<br>0x4 = CMOS Output mode and power supply of 3.3V.<br>0x5 = Reserved<br>0x6 = CMOS Output mode and power supply of 1.8V.<br>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<br>0x0 = Open drain Output mode. Standard mode.<br>0x1 = Open drain Output mode. Fast mode.<br>0x2 = Reserved<br>0x3 = Open drain Output mode. Fast mode plus.<br>0x4 = CMOS Output mode and power supply of 3.3V.<br>0x5 = Reserved<br>0x6 = CMOS Output mode and power supply of 1.8V.<br>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<br>0x0 = Open drain Output mode. Standard mode.<br>0x1 = Open drain Output mode. Fast mode.<br>0x2 = Reserved<br>0x3 = Open drain Output mode. Fast mode plus.<br>0x4 = CMOS Output mode and power supply of 3.3V.<br>0x5 = Reserved<br>0x6 = CMOS Output mode and power supply of 1.8V.<br>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 tuning cap select<br>0x0 = 5.5pF<br>0x1 = 5.5pF + 504fF<br>0x2 = 5.5pF + 1.008pF<br>0x3 = 5.5pF + 1.512pF<br>0x4 = 5.5pF + 2.016pF<br>0x5 = 5.5pF + 2.520pF<br>0x6 = 5.5pF + 3.024pF<br>0x7 = 5.5pF + 3.528pF<br>0x8 = 5.5pF + 4.032pF<br>0x9 = 5.5pF + 4.536pF |
|                                |                  |            |               | xobuf_digicap_x2 (continued)  |
| 11:8                           | xobuf_digicap_x1 | RW         | 0x0           | XIN internal tuning cap select<br>0x0 = 5.5pF<br>0x1 = 5.5pF + 504fF<br>0x2 = 5.5pF + 1.008pF<br>0x3 = 5.5pF + 1.512pF<br>0x4 = 5.5pF + 2.016pF<br>0x5 = 5.5pF + 2.520pF<br>0x6 = 5.5pF + 3.024pF<br>0x7 = 5.5pF + 3.528pF<br>0x8 = 5.5pF + 4.032pF<br>0x9 = 5.5pF + 4.536pF  |
|                                |                  |            |               | xobuf_digicap_x1 (continued)  |
| 7:0                            | reserved         | RO         | 0x3           | Reserved  |

### 3.5 SYSDIV

System Clock Divider Registers.

Table 9. SYSDIV Register Index

| Offset (Hex) | Register Module Base Address: 0x94 |                                    |
|--------------|------------------------------------|------------------------------------|
|              | Register Name                      | Register Description               |
| 0x0          | <a href="#">SYS_DIV_INT_CNFG</a>   | System Clock Divider Configuration |
| 0x1          | <a href="#">COUNTER_1US_CNFG</a>   | System 1 us Counter Configuration  |
| 0x2          | <a href="#">SYS_DIV_EN_CTRL</a>    | System Clock Divider Enable        |

#### 3.5.1 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. |

#### 3.5.2 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. |

#### 3.5.3 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<br>0x1 = Enable<br>0x0 = Disable |

### 3.6 GPIO

General Purpose IO Registers.

Instances 0-7 apply to GPIO0 to GPIO9.

Instance 8 applies to LOCK.

Table 10. 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                   |

#### 3.6.1 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.<br>When set, will send out a synchronized version of the input and may have a sampling variation of 1 to 2 system clocks<br>0x0 = Deglitcher in use<br>0x1 = Deglitcher bypassed  |
| 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.<br>0x0 = No synchronization<br>0x1 = Synchronization to the system clock is enabled |
| 12                               | gpio_pol             | RW         | 0x0           | GPIO polarity. Inverts input or output<br>0x0 = Do not invert<br>0x1 = Invert   |
| 11                               | reserved             | RO         | 0x0           | Reserved  |
| 10:8                             | gpio_startup_mode    | RW         | 0x1           | GPIO startup function. Chooses startup function<br>0x0 = config_index[0]<br>0x1 = config_index[1]<br>0x2 = I2C_addr[0]<br>0x3 = I2C_addr[1]<br>0x4 = I2C_addr[2]<br>0x5 = Reserved<br>0x6 = Reserved<br>0x7 = Reserved  |

| GPIO_CNFG Bit Field Descriptions |                       |            |   |  |
|----------------------------------|-----------------------|------------|---|--|
| Bit Field                        | Field Name            | Field Type | Default Value                           | Description  |
| 7:0                              | gpio_func             | RW         | 0x20,0x20,0x20,0x20,0x20,0x20,0x20,0x1B | GPIO function. Select the function of the corresponding GPIO.<br>For the output functions, the corresponding pad_gpio_oe_b bit must be set to 0.<br>0x0 = Logic Low (output)<br>0x1 = Logic High (output)<br>0x2 = XIN LOS (output)<br>0x3 = Reserved<br>0x4 = Reserved<br>0x5 = Reserved<br>0x6 = Reserved<br>0x8 = Reserved<br>0x9 = Reserved<br>0xA = Reserved  |
|                                  | gpio_func (continued) |            |   | 0xB = Reserved<br>0xC = Reserved<br>0xD = Reserved<br>0xE = Reserved<br>0xF = Reserved<br>0x11 = XIN LOS limit (output)<br>0x12 = Reserved<br>0x13 = Reserved<br>0x14 = Reserved<br>0x15 = Reserved  |
|                                  | gpio_func (continued) |            |   | 0x17 = Load complete (OTP or EEPROM loaded) (output)<br>0x18 = Device Ready (startup sequence complete) (output)<br>0x19 = APLL rail low (output)<br>0x1A = APLL rail high (output)<br>0x1B = APLL lock (from frequency-based lock detect) (output)<br>0x1C = APLL lol (output)<br>0x1D = APLL lol_lmt (output)<br>0x1E = APLL LF lock (VCTRL-based lock) (output)<br>0x1F = FCL lock (aux loop status) (output)<br>0x20 = General purpose input (input) |
|                                  | gpio_func (continued) |            |   | 0x30 = Reserved<br>0x31 = Reserved<br>0x32 = Reserved<br>0x33 = Reserved<br>0x34 = Reserved<br>0x35 = Reserved<br>0x3C = Reserved<br>0x4C = Global OE (input)<br>0x4D = Group OE[0:3] (input)<br>0x4E = Group OE[4:7] (input)  |
|                                  | gpio_func (continued) |            |   | 0x4F = Group OE[8:11] (input)<br>0x50 = OE[0] (input)<br>0x51 = OE[1] (input)<br>0x52 = OE[2] (input)<br>0x53 = OE[3] (input)<br>0x54 = OE[4] (input)<br>0x55 = OE[5] (input)<br>0x56 = OE[6] (input)<br>0x57 = OE[7] (input)<br>0x58 = OE[8] (input)  |

| GPIO_CNFG Bit Field Descriptions |                       |            |               |  |
|----------------------------------|-----------------------|------------|---------------|--|
| Bit Field                        | Field Name            | Field Type | Default Value | Description  |
|                                  | gpio_func (continued) |            |               | 0x59 = OE[9] (input)<br>0x5A = OE[10] (input)<br>0x5B = OE[11] (input)<br>0x5C = Reserved<br>0x5D = Reserved<br>0x5E = Reserved<br>0x5F = Reserved<br>0x60 = Reserved<br>0x61 = Reserved<br>0x62 = Reserved  |
|                                  | gpio_func (continued) |            |               | 0x63 = Reserved<br>0x6C = Reserved<br>0x6D = Reserved<br>0x6E = Reserved<br>0x6F = Reserved<br>0x7F = Device Global Interrupt (output)<br>0x80 = Device interrupt[0] - APLL LOL (frequency-based lock detector) (output)<br>0x81 = Device interrupt[1] - APLL Lock (frequency-based lock detector) (output)<br>0x82 = Device interrupt[2] - APLL LF valid (VCTRL-based lock) (output)<br>0x83 = Device interrupt[3] - APLL FCL lock (aux loop status) (output) |
|                                  | gpio_func (continued) |            |               | 0x84 = Reserved<br>0x86 = Reserved<br>0x88 = Reserved<br>0x8A = Reserved<br>0x8B = Reserved<br>0x8C = Reserved<br>0x8D = Reserved<br>0x8E = Device interrupt[14] - XIN0 LOS (output)<br>0x90 = Reserved<br>0x91 = Reserved   |
|                                  | gpio_func (continued) |            |               | 0x92 = Reserved<br>0x93 = Reserved<br>0x94 = Reserved<br>0x95 = Reserved<br>0x96 = Reserved<br>0x97 = Reserved<br>0x98 = Device interrupt[24] - APLL LOL LMT (output)<br>0x99 = Reserved<br>0x9B = Reserved<br>0x9C = Reserved   |
|                                  | gpio_func (continued) |            |               | 0x9D = Reserved<br>0x9E = Reserved<br>0x9F = Device interrupt[31] - XIN0 LOS LMT (output)<br>0xA1 = Device interrupt[33] - OTP MANUAL RDY (output)<br>0xA2 = Device interrupt[34] - OTP/EEPROM CRC ERR (output)<br>0xA3 = Device interrupt[35] - OTP/EEPROM FAIL EVT (output)<br>0xA4 = Reserved<br>0xA5 = Reserved<br>0xA6 = Reserved<br>0xA7 = Reserved  |

| GPIO_CNFG Bit Field Descriptions |                       |            |               |  |
|----------------------------------|-----------------------|------------|---------------|--|
| Bit Field                        | Field Name            | Field Type | Default Value | Description  |
|                                  | gpio_func (continued) |            |               | 0xA9 = Device interrupt[38] - I2C CRC ERR (output)<br>0xD4 = clk_cali_div8 (output) (set gpio_resync to 0)<br>0xD5 = clk_xin (output) (set gpio_resync to 0)<br>0xD6 = clk_dcd (output) (set gpio_resync to 0)<br>0xD7 = Reserved<br>0xD8 = Reserved<br>0xD9 = Reserved<br>0xDA = Reserved<br>0xDB = Reserved<br>0xDE = Reserved |
|                                  | gpio_func (continued) |            |               | 0xDF = Reserved<br>0xE0 = Reserved<br>0xE1 = Reserved<br>0xE3 = clk_apll_test_mux (output) (set gpio_resync to 0)  |

### 3.6.2 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 |

### 3.6.3 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)<br>0x0 = Enable<br>0x1 = Disable  |
| 4                                    | pad_gpio_ru_b | RW         | 0x0                                     | Pull up enable. Pull up enable (active low)<br>0x0 = Enable<br>0x1 = Disable  |
| 3                                    | pad_gpio_oe_b | RW         | 0x1,0x1,<br>0x1,0x1,<br>0x1,0x1,<br>0x0 | Output enable. Enable output buffer (active low). The output buffer must be disabled when gpio_func is configured for an input function.<br>0x0 = Enable<br>0x1 = Disable   |
| 2:0                                  | pad_gpio_drv  | RW         | 0x5                                     | Drive strength. Drive Strength<br>0x0 = Open drain Output mode. Standard mode.<br>0x1 = Open drain Output mode. Fast mode.<br>0x2 = Open drain Output mode. Not used<br>0x3 = Open drain Output mode. Fast mode plus.<br>0x4 = Reserved<br>0x5 = Reserved<br>0x6 = CMOS Output mode and power supply of 1.8V.<br>0x7 = Reserved |

### 3.6.4 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. |

### 3.7 OUTBUF

Output Buffer Registers.

Table 11. OUTBUF Register Index

| Offset (Hex) | Register Module Base Address: 0x100 |                                       |
|--------------|-------------------------------------|---------------------------------------|
|              | Register Name                       | Register Description                  |
| 0x0          | <a href="#">OUT_CNFG</a>            | Output Buffer General Configuration   |
| 0x1          | <a href="#">OUT_MODE_CNFG</a>       | Output Buffer Mode Configuration      |
| 0x2          | <a href="#">OUT_HCSL_CMOS_CNFG</a>  | Output Buffer HCSL CMOS Configuration |
| 0x3          | <a href="#">OUT_LVDS_CNFG</a>       | Output Buffer LVDS Configuration      |
| 0x4          | <a href="#">OUT_EN_CTRL</a>         | Output Pad Enables                    |
| 0x6          | <a href="#">OUT_SPARE</a>           | Spare Output Buffer Bits              |

#### 3.7.1 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.<br>0x0 = Disable<br>0x1 = Enable |
| 3:0                             | reserved    | RW         | 0x8           | Reserved   |

#### 3.7.2 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    |

| OUT_MODE_CNFG Bit Field Descriptions |               |            |               |  |
|--------------------------------------|---------------|------------|---------------|--|
| Bit Field                            | Field Name    | Field Type | Default Value | Description  |
| 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.<br>0x0 = Held Low / Low (except LVDS/HCSL mode is held Low / High).<br>0x1 = Held Low / High.<br>0x2 = Held Hi-Z / Hi-Z.<br>0x3 = Output power down (regardless of output enable of the driver) |
| 1:0                                  | out_mode      | RW         | 0x3           | Output Driver type. Output Driver type.<br>0x0 = HCSL<br>0x1 = Reserved<br>0x2 = LVDS<br>0x3 = CMOS  |

### 3.7.3 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.<br>0x0 = OUTx, nOUTx are driven with the same phase.<br>0x1 = OUTx, nOUTx are driven with the opposite phase.<br>0x2 = Only OUTx is driven. nOUTx is held low.<br>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.<br>0x0 = Internal HCSL termination is disabled. An external termination resistor of 50 ohms to ground is required.<br>0x1 = Internal HCSL termination is enabled, providing an internal 50 ohm resistor to ground.                               |
| 3:0                                       | out_cnf_hcsl_swing             | RW         | 0x0           | Output Driver HCSL amplitude control. Controls the amplitude of the output driver when HCSL mode is selected. Each value increment provides an approximate 50mV increase.<br>0x0 = 213mV<br>0x1 = 266mV<br>0x2 = 321mV<br>0x3 = 374mV<br>0x4 = 430mV<br>0x5 = 482mV<br>0x6 = 537mV<br>0x7 = 589mV<br>0x8 = 641mV<br>0x9 = 692mV |
|   | out_cnf_hcsl_swing (continued) |            |               | 0xA = 742mV<br>0xB = 789mV<br>0xC = 834mV<br>0xD = 875mV<br>0xE = 913mV<br>0xF = 946mV  |

### 3.7.4 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.<br>0x0 = 377mV<br>0x1 = 456mV  |
| 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.<br>0x0 = Reserved<br>0x1 = 900mV<br>0x2 = 1000mV<br>0x3 = 1100mV |

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

### 3.7.6 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           | Boosts HCSL swing from 946mV to 1085mV.<br>Note: Available for out_cnf_hcsl_swing = 0xF only.<br>out_spare[0] bits <7:0> correspond to en_hcsl_boost[7:0]<br>out_spare[1] bits <3:0> correspond to en_hcsl_boost[11:8]<br>0 = Disable HCSL amplitude boost<br>1 = Enable HCSL amplitude boost |

### 3.8 TDCAPLL

TDC APLL Registers.

Table 12. TDCAPLL Register Index

| Offset (Hex) | Register Module Base Address: 0x160 |  |
|--------------|-------------------------------------|--|
|              | Register Name                       | Register Description                       |
| 0x2          | <a href="#">TDC_FB_DIV_INT_CNFG</a> | TDC Feedback Divider Integer Configuration |
| 0x3          | <a href="#">TDC_REF_DIV_CNFG</a>    | TDC Reference Divider Configuration        |
| 0x9          | <a href="#">TDC_ENABLE_CTRL</a>     | TDC Filter Status                          |

#### 3.8.1 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. |

#### 3.8.2 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.<br>0x0 = Bypass divider.<br>0x1 = Divide by 2<br>0x2 = Divide by 4<br>0x3 = Divide by 8<br>0x4 = Divide by 16 |

### 3.8.3 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. Power consumption can be reduced by disabling the TDC when it is not needed.<br>0x0 = Disabled<br>0x1 = Enabled |

## 3.9 XTALMON

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

Table 13. XTALMON Register Index

| Offset (Hex) | Register Module Base Address: 0x170         |   |
|--------------|---|---|
|              | Register Name                               | Register Description                          |
| 0x0          | <a href="#">XTALMON_NOMINAL_MARGIN_CNFG</a> | XTAL Monitor Nominal and Margin Configuration |
| 0x4          | <a href="#">XTALMON_WINDOW_CNFG</a>         | XTAL Monitor Miscellaneous Configuration      |
| 0x6          | <a href="#">XTALMON_QUAL_CNFG</a>           | XTAL Monitor Qualification Configuration      |
| 0x7          | <a href="#">XTALMON_CTRL</a>                | XTAL Monitor enable                           |
| 0x8          | <a href="#">XTALMON_EVENT</a>               | XTAL Monitor Events                           |
| 0x9          | <a href="#">XTALMON_CNT_EVENT</a>           | XTAL Counter Status                           |
| 0xA          | <a href="#">XTALMON_STS</a>                 | XTAL Monitor Status                           |

### 3.9.1 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.   |

### 3.9.2 XTALMON\_WINDOW\_CNFG

XTAL Monitor Miscellaneous Configuration.

| XTALMON_WINDOW_CNFG Bit Field Descriptions |                     |            |               |  |
|--|---------------------|------------|---------------|--|
| Bit Field                                  | Field Name          | Field Type | Default Value | Description  |
| 15:12                                      | reserved            | RO         | 0x0           | Reserved   |
| 11:8                                       | xtal_los_cnt_thresh | RW         | 0x0           | Loss-of-Signal Counter Threshold. While the Loss-of-Signal counter (los_cnt) exceeds this threshold, the xtal_los_lmt_evt bit is set. The maximum valid value for this field is 30 (0x1E).   |
| 7:5  | 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. |

### 3.9.3 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.    |

### 3.9.4 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.<br>0x0 = XTAL LOS monitor disabled<br>0x1 = XTAL LOS monitor enabled |

### 3.9.5 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).<br>0x0 = Loss-of-signal counter has not exceeded the threshold since the last time the bit was cleared<br>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.<br>0x0 = Loss-of-signal not detected since the last time the bit was cleared<br>0x1 = Loss-of-signal detected since the last time the bit was cleared  |

### 3.9.6 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. |

### 3.9.7 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:<br>0x0 = Clock meets the monitoring criteria<br>0x1 = Loss-of-signal detected |

### 3.10 APLL

APLL Registers.

Table 14. APLL Register Index

| Offset (Hex) | Register Module Base Address: 0x240   |                                     |
|--------------|---------------------------------------|-------------------------------------|
|              | Register Name                         | Register Description                |
| 0x0          | <a href="#">LPF_TC_CNFG</a>           | VCO Temp Compensation Configuration |
| 0x10         | <a href="#">APLL_FB_DIV_FRAC_CNFG</a> | APLL Feedback Divider Fraction      |

Table 14. APLL Register Index

| Offset (Hex) | Register Module Base Address: 0x240  |                                    |
|--------------|--------------------------------------|------------------------------------|
|              | Register Name                        | Register Description               |
| 0x18         | <a href="#">APLL_FB_DIV_INT_CNFG</a> | APLL Feedback Divider Integer      |
| 0x1A         | <a href="#">APLL_DCD_CAL_CNFG</a>    | APLL DCD Calibration Configuration |

### 3.10.1 LPF\_TC\_CNFG

VCO Temp Compensation Configuration.

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

### 3.10.2 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         | 0x1CD5<br>555555 | APLL Feedback Divider Fraction. APLL feedback divider numerator value. The denominator is a fixed value of 2 <sup>38</sup> . |

### 3.10.3 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. |

### 3.10.4 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    |

| APLL_DCD_CAL_CNFG Bit Field Descriptions |              |            |               |   |
|--|--------------|------------|---------------|---|
| Bit Field                                | Field Name   | Field Type | Default Value | Description   |
| 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  |

### 3.11 IOD

Integer Output Divider Registers.

Table 15. IOD Register Index

| Offset (Hex) | Register Module Base Address: 0x2A0 |  |
|--------------|-------------------------------------|--|
|              | Register Name                       | Register Description   |
| 0x0          | <a href="#">IOD_DIV_CNFG</a>        | Integer Output Divider Ratio Configuration Register            |
| 0x4          | <a href="#">IOD_PHASE_CNFG</a>      | Integer Output Divider Phase Adjustment Configuration Register |
| 0x6          | <a href="#">IOD_CNFG</a>            | Integer Output Divider Configuration Register                  |
| 0x7          | <a href="#">IOD_EN_CTRL</a>         | Integer Output Divider Enable Register                         |
| 0x8          | <a href="#">IOD_PHASE_EN_CTRL</a>   | Integer Output Divider Phase Adjust Trigger                    |

#### 3.11.1 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. |

#### 3.11.2 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.<br>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. |

### 3.11.3 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.<br>0x0 = Do not apply after a sync event<br>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.<br>For IOD[3:0] and IOD[11:8], when set, this bit enables the APLL VCO fanout buffer to the IOD selection mux.<br>0x0 = Disable<br>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.<br>0x0 = VCO/2<br>0x1 = FOD0<br>0x2 = FOD1<br>0x3 = FOD2<br>0x4 = Reserved<br>0x5 = Reserved<br>0x6 = Reserved<br>0x7 = Static high (Minimal power)  |

### 3.11.4 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 | IOD Enable. This bit enables or disables the IOD.<br>0x0 = Disabled<br>0x1 = Enabled |

### 3.11.5 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.<br>This bit should not be set to one if the IOD is disabled (if the corresponding out_driver_en bit is low). |

### 3.12 FOD

Fractional Output Divider Registers.

Table 16. 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                      |

#### 3.12.1 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.<br>0x0 = Sync not applied<br>0x1 = Sync is applied  |
| 3:2                             | reserved             | RO         | 0x1           | Reserved   |
| 1                               | fod_sync_mode        | RW         | 0x0           | FOD synchronous mode enable. Enables the synchronous mode of the FOD<br>See fod_div_integer for the formulas<br>0x0 = Non-synchronous mode, fod_div_numerator and fod_div_denominator are not used.<br>0x1 = Synchronous mode, fod_div_numerator and fod_div_denominator are used. |
| 0                               | reserved             | RW         | 0x0           | Reserved   |

#### 3.12.2 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).<br>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. |

### 3.12.3 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. |

### 3.12.4 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         | 0x80000000    | FOD denominator. FOD denominator for Synchronous mode (fod_sync_mode=1) only. See fod_div_integer for the formula. |

### 3.12.5 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.<br>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.<br>if fod_sync_mode=0: FOD divide ratio = fod_div_integer + fod_div_fraction[39:0]/2 <sup>40</sup><br>if fod_sync_mode=1: FOD divide ratio = fod_div_integer + (fod_div_fraction[15:0] + fod_div_numerator/fod_div_denominator)/2 <sup>16</sup> |
| 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   |

### 3.12.6 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<br>0x0 = Disable & reset FOD<br>0x1 = Enable |

### 3.12.7 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. |

### 3.12.8 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 <sup>-44</sup> , range ±244 ppm |

## 3.13 LDO

LDO Registers.

Table 17. LDO Register Index

| Offset (Hex) | Register Module Base Address: 0x4C0 |                       |
|--------------|-------------------------------------|-----------------------|
|              | Register Name                       | Register Description  |
| 0x24         | <a href="#">FOD_LDO_CNFG</a>        | FOD LDO Configuration |

### 3.13.1 FOD\_LDO\_CNFG

FOD LDO Configuration.

| FOD_LDO_CNFG Bit Field Descriptions |            |            |               |  |
|-------------------------------------|------------|------------|---------------|--|
| Bit Field                           | Field Name | Field Type | Default Value | Description  |
| 7:1                                 | reserved   | RO         | 0x28          | Reserved   |
| 0                                   | fod_en_ldo | RW         | 0x1           | FOD LDO enable. FOD LDO enable.<br>0x0 = Disable<br>0x1 = Enable |

## 3.14 DPLL

DPLL Registers.

Table 18. DPLL Register Index

| Offset (Hex) | Register Module Base Address: 0x500   |                                |
|--------------|---------------------------------------|--------------------------------|
|              | Register Name                         | Register Description           |
| 0x4          | <a href="#">DPLL_MODE_CNFG</a>        | DPLL Mode Configuration        |
| 0x6          | <a href="#">DPLL_XTAL_OFFSET_CNFG</a> | DPLL XTAL Offset Configuration |
| 0x60         | <a href="#">DPLL_WR_FREQ_CTRL</a>     | DPLL Write Frequency Control   |

### 3.14.1 DPLL\_MODE\_CNFG

DPLL Mode Configuration.

| DPLL_MODE_CNFG Bit Field Descriptions |            |            |               |  |
|---------------------------------------|------------|------------|---------------|--|
| Bit Field                             | Field Name | Field Type | Default Value | Description  |
| 15:3                                  | reserved   | RW         | 0x100c        | Reserved   |
| 2:0                                   | dpll_mode  | RW         | 0x6           | DPLL mode selection. Selects the mode or state for the DCO.<br>0x0 = Freerun<br>0x1 = Reserved<br>0x2 = Reserved<br>0x3 = Write Frequency<br>0x4 = Reserved<br>0x5 = Reserved<br>0x6 = Freerun<br>0x7 = Reserved |

### 3.14.2 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 = $\pm 2^{-13}$ (~ $\pm 122$ ppm). <code>apll_fb_sdm_order</code> must be set to a value greater than 0 for <code>xtal_trim</code> to operate correctly. |

### 3.14.3 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 $\pm 243$ ppm. <code>apll_fb_sdm_order</code> must be set to a value greater than 0 for <code>write_freq</code> 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. |

## 3.15 EEPROM

EEPROM Registers.

Table 19. EEPROM Register Index

| Offset (Hex) | Register Module Base Address: 0x600  |                              |
|--------------|--------------------------------------|------------------------------|
|              | Register Name                        | Register Description         |
| 0x0          | <a href="#">EEPROM_CNFG</a>          | EEPROM Configuration         |
| 0x8          | <a href="#">EEPROM_ADDR_CNFG</a>     | EEPROM Address Configuration |
| 0x9          | <a href="#">EEPROM_EVENT</a>         | EEPROM Events                |
| 0xA          | <a href="#">EEPROM_ERR_CNT_EVENT</a> | EEPROM Error Counter Status  |

### 3.15.1 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                              | eeeprom_fall      | RW         | 0x96          | EEPROM falling edge time. Cycle number (counting down from eeeprom_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                              | eeeprom_rise      | RW         | 0x1C2         | EEPROM rising edge time. Cycle number (counting down from eeeprom_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                              | eeeprom_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                                 | eeeprom_ext_addr  | RW         | 0x0           | EEPROM extended address enable. Allows extended 10-bit addressing with a 1-byte I2C address, if supported by the EEPROM.<br>0x0 = The address is sent outside of the device address.<br>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. eeeprom_addr_size must be set to 0.   |
| 11                                 | eeeprom_addr_size | RW         | 0x1           | EEPROM address size. Number of address bytes sent to the EEPROM during a read.<br>0x0 = 1-byte address<br>0x1 = 2-byte address   |
| 10:7                               | eeeprom_length    | RW         | 0x4           | EEPROM size. Selects the number of bytes in the EEPROM for storing configurations.<br>0x0 = 128B<br>0x1 = 256B<br>0x2 = 512B<br>0x3 = 1KB<br>0x4 = 2KB<br>0x5 = 4KB<br>0x6 = 8KB<br>0x7 = 16KB<br>0x8 = 32KB<br>0x9 = 64KB   |
| 6                                  | eeeprom_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 eeeprom_load_en is set to 1), OTP user, and EEPROM user (if eeeprom_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.<br>0x0 = Disabled<br>0x1 = Enabled |

| EEPROM_CNFG Bit Field Descriptions |                     |            |               |   |
|------------------------------------|---------------------|------------|---------------|---|
| Bit Field                          | Field Name          | Field Type | Default Value | Description   |
| 5:4                                | eeeprom_i2c_speed   | RW         | 0x0           | EEPROM I2C speed. Selects the I2C master speed for EEPROM load. When the speed is 400kHz or 1MHz, eeeprom_fall, eeeprom_rise and eeeprom_cycle are internally divided by 4 or 10 respectively to achieve the faster timing. The pad drive strength (eeeprom_i2c_drv) should also be set according to the speed.<br>0x0 = 100kHz<br>0x1 = 400kHz<br>0x2 = 1MHz<br>0x3 = Reserved |
| 3:0                                | eeeprom_retry_count | RW         | 0x4           | EEPROM Load Retry Count. Number of times to attempt to load EEPROM. If eeeprom_bad or load failed (CRC error) after this number of attempts, the eeeprom_load_fail status bit is set.   |

### 3.15.2 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                                     | eeeprom_addr | RW         | 0x50          | EEPROM Device Address. Sets the I2C device address of the EEPROM to load. (R.3.1.13.20) |

### 3.15.3 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                                   | eeeprom_bad_evt          | RW1C       | 0x0           | EEPROM Not Detected. When high, indicates the EEPROM did not acknowledge a read access during device startup. In this case, eeeprom_load_fail is not set. If EEPROM load is disabled (eeeprom_load_en is set to 0), then the chip will not attempt to read the EEPROM and this bit cannot be set. Cleared by writing it to 1. |
| 2                                   | eeeprom_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                                   | eeeprom_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 (eeeprom_bad is set instead). Cleared by writing it to 1.  |
| 0                                   | eeeprom_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.   |

### 3.15.4 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   | eeeprom_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   |
|----------|--------------|---|
| 1.03     | Dec 19, 2024 | Changed the maximum time delay number in <a href="#">Clock Output Enable</a> to 14ns from 13ns  |
| 1.02     | Dec 16, 2024 | <ul style="list-style-type: none"> <li>▪ Added <a href="#">Clock Output Enable</a>.</li> <li>▪ Updated <a href="#">pad_gpio_drv</a>.</li> </ul>   |
| 1.01     | Nov 21, 2024 | <ul style="list-style-type: none"> <li>▪ Updated the following Functional Description items: <a href="#">Figure 3, Fractional Output Dividers</a> (introductory paragraphs), <a href="#">FOD Synchronous Mode</a>, <a href="#">Crystal Monitor</a>, <a href="#">Table 1</a>, <a href="#">Figure 5</a>.</li> <li>▪ Updated the third paragraph of <a href="#">I2C Slave</a>.</li> <li>▪ Updated the following Registers items: <a href="#">xobuf_digicap_x2</a>, <a href="#">xobuf_digicap_x1</a>, <a href="#">gpio_deglitch_bypass</a>, <a href="#">gpio_resync</a>, <a href="#">gpio_func</a>, <a href="#">pad_gpio_oe_b</a>, <a href="#">pad_gpio_drv</a>, <a href="#">out_cnf_hcsl_swing</a>, <a href="#">out_cnf_lvds_amp</a>, <a href="#">out_lvds_cm_voltage</a>, <a href="#">out_spare</a>, <a href="#">TDC_ENABLE_CTRL</a>.</li> <li>▪ Added the following Registers items: <a href="#">LPF_TC_CNFG</a> and <a href="#">FOD_LDO_CNFG</a>.</li> <li>▪ Removed the following Registers items: <a href="#">APLL_DCD_SDM_CNFG</a>, <a href="#">dpll_bw_set_int_en</a>, and <a href="#">fod_integer_mode</a>.</li> </ul> |
| 1.00     | Jun 10, 2024 | Initial release.  |

## IMPORTANT NOTICE AND DISCLAIMER

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

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

(Disclaimer Rev.1.01)

### Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,  
Koto-ku, Tokyo 135-0061, Japan  
[www.renesas.com](http://www.renesas.com)

### Trademarks

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

### Contact Information

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