



Features

• Low-Bandwidth DPLL

- ITU-T G.813/G.8262 compliance (options 1 & 2)
- Low-jitter operation from any ≥ 10 MHz TCXO
- Master clock jitter attenuator reduces cost by removing TCXO/OCXO low-jitter requirement
- Programmable bandwidth, 0.1Hz to 10Hz
- Attenuates input clock jitter up to several UI
- Hitless reference switching
- High-resolution holdover averaging
- Digitally controlled phase adjustment

• Input Clocks

- Up to 3 inputs, 2 differential/CMOS, one CMOS
- Any input frequency from 8kHz to 1250MHz (8kHz to 300MHz for CMOS)
- Per-input activity and frequency monitoring
- Automatic or manual reference switching

• Low-Jitter Fractional-N APLL and 3 Outputs

- Any output frequency from <1Hz to 1035MHz
- High-resolution fractional frequency conversion with 0ppm error
- Encapsulated design requires no external VCXO or loop filter components
- Each output has independent dividers
- Output jitter as low as 0.25ps RMS (12kHz-20MHz integration band)

Ordering Information

ZL30621LFG7	64 Pin LGA	Trays
ZL30621LFF7	64 Pin LGA	Tape and Reel
Ni Au		
Package size: 5 x 10 mm		

-40°C to +85°C

- Outputs are CML or 2xCMOS, can interface to LVDS, LVPECL, HSTL, SSTL and HCSL
- In 2xCMOS mode, the P and N pins can be different frequencies (e.g. 125MHz and 25MHz)
- Per-output supply pin with CMOS output voltages from 1.5V to 3.3V
- Precise output alignment circuitry and per-output phase adjustment
- Per-output enable/disable and glitchless start/stop (stop high or low)

• General Features

- Automatic self-configuration at power-up from internal EEPROM; up to four configurations pin-selectable
- Numerically controlled oscillator mode
- Input-to-output alignment with external feedback
- SPI or I²C processor Interface
- Easy-to-use evaluation software

Applications

- Telecom timing cards for SONET/SDH, SyncE, wireless base stations and other systems

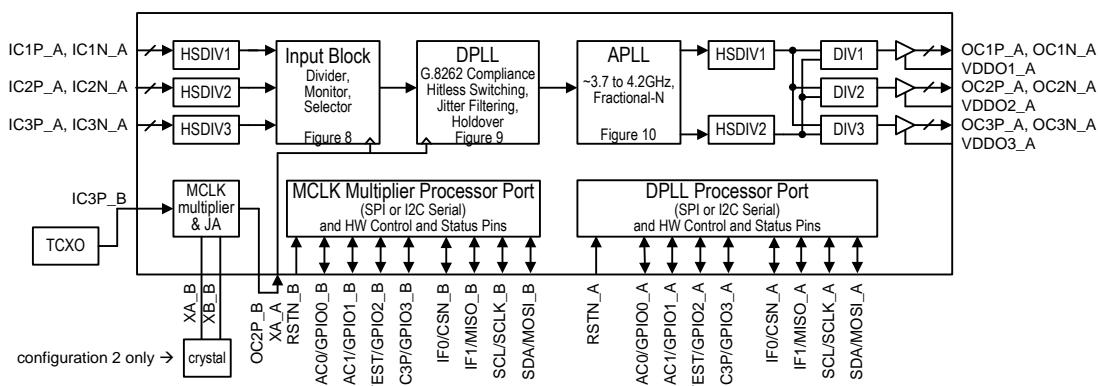


Table of Contents

1. APPLICATION EXAMPLES	5
2. DETAILED FEATURES	5
2.1 MASTER CLOCK JITTER ATTENUATOR AND MULTIPLIER	5
2.2 INPUT BLOCK FEATURES	5
2.3 DPLL FEATURES	5
2.4 APLL FEATURES	6
2.5 OUTPUT CLOCK FEATURES	6
2.6 GENERAL FEATURES	6
2.7 EVALUATION SOFTWARE	6
3. PIN DIAGRAM	7
4. PIN DESCRIPTIONS	8
5. FUNCTIONAL DESCRIPTION	10
5.1 OVERVIEW: TWO MACRO-BLOCKS	10
5.2 DEVICE IDENTIFICATION	10
5.3 TOP-LEVEL CONFIGURATION	10
5.3.1 Configurations 1 & 2: Block B as Master Clock Multiplier and/or Jitter Attenuator	10
5.3.2 Configuration 3: Block B as Output Frequency Converter and Jitter Attenuator	11
5.3.3 Configuration Comparison	12
5.3.4 Evaluation Software for Device Configuration	12
5.4 PIN-CONTROLLED AUTOMATIC CONFIGURATION AT RESET	12
5.5 LOCAL OSCILLATOR OR CRYSTAL	13
5.5.1 External Oscillator	13
5.5.2 External Crystal and On-Chip Driver Circuit	13
5.5.3 Ring Oscillator (for System Start-Up)	14
5.6 INPUT SIGNAL FORMAT CONFIGURATION	14
5.7 INPUT BLOCK: INPUT DIVIDER, MONITOR AND SELECTOR	15
5.7.1 Input Clock Inversion and High-Speed Dividers	15
5.7.2 Input Clock Monitoring	15
5.7.3 Input Clock Priority, Selection and Switching for the DPLL	17
5.8 DPLL ARCHITECTURE AND CONFIGURATION	18
5.8.1 DPLL Configuration	18
5.8.2 DPLL States	19
5.8.3 DPLL Capabilities	19
5.8.4 Input Wander and Jitter Tolerance	20
5.8.5 Jitter and Wander Transfer	20
5.8.6 Output Jitter and Wander	20
5.8.7 Numerically Controlled Oscillator (NCO) Mode	21
5.9 APLL CONFIGURATION	21
5.9.1 APLL Signal Source	21
5.9.2 APLL Output Frequency	22
5.9.3 APLL Phase Adjustment	23
5.10 OUTPUT CLOCK CONFIGURATION	23
5.10.1 Output Enable, Signal Format, Voltage and Interfacing	23
5.10.2 Output Frequency Configuration	23
5.10.3 Output Duty Cycle Adjustment	24
5.10.4 Output Phase Adjustment and Phase Alignment	25
5.10.5 Output Clock Start and Stop	27
5.10.6 A-to-B Phase Offset Measurement	28

5.11	MICROPROCESSOR INTERFACE	30
5.11.1	<i>SPI Slave</i>	30
5.11.2	<i>I²C Slave</i>	32
5.12	INTERRUPT LOGIC	34
5.13	RESET LOGIC	35
5.14	POWER-SUPPLY CONSIDERATIONS	35
5.15	AUTO-CONFIGURATION FROM EEPROM	35
5.15.1	<i>Generating Device Configurations</i>	35
5.15.2	<i>Direct EEPROM Write Mode</i>	35
5.15.3	<i>Holding Other Devices in Reset During Auto-Configuration</i>	36
5.16	POWER SUPPLY DECOUPLING AND LAYOUT RECOMMENDATIONS	36
6.	REGISTER DESCRIPTIONS	36
6.1	REGISTER TYPES	36
6.1.1	<i>Status Bits</i>	36
6.1.2	<i>Configuration Fields</i>	36
6.1.3	<i>Multiregister Fields</i>	36
6.1.4	<i>Bank-Switched Registers</i>	37
6.1.5	<i>DPLL Registers</i>	37
6.2	REGISTER MAP	37
6.3	REGISTER DEFINITIONS	40
6.3.1	<i>Global Configuration Registers</i>	40
6.3.2	<i>Status Registers</i>	49
6.3.3	<i>APLL Configuration Registers</i>	65
6.3.4	<i>Output Clock Configuration Registers</i>	71
6.3.5	<i>Input Clock Configuration Registers</i>	77
6.3.6	<i>DPLL Configuration Registers</i>	78
7.	ELECTRICAL CHARACTERISTICS	84
8.	PACKAGE AND THERMAL INFORMATION	94
8.1	PACKAGE TOP MARK FORMAT	94
8.2	THERMAL SPECIFICATIONS	95
9.	MECHANICAL DRAWING	96
10.	ACRONYMS AND ABBREVIATIONS	97
11.	DATA SHEET REVISION HISTORY	98

List of Figures

Figure 1 - Functional Block Diagram	1
Figure 2 – Telecom Timing Card Application, TCXO + Crystal/XO for Lowest Jitter	5
Figure 3 – Telecom Timing Card Application, TCXO Only	5
Figure 4 - Pin Diagram.....	7
Figure 5 – Top-Level Configurations 1 and 2	11
Figure 6 – Top-Level Configuration 3	11
Figure 7 - Crystal Equivalent Circuit / Recommended Crystal Circuit	14
Figure 8 - Input Block Diagram	15
Figure 9 - DPLL Block Diagram	18
Figure 10 - APLL Block Diagram	22
Figure 11 - SPI Read Transaction Functional Timing.....	31
Figure 12 - SPI Write Enable Transaction Functional Timing	31
Figure 13 - SPI Write Transaction Functional Timing.....	32
Figure 14 – I ² C Read Transaction Functional Timing.....	33
Figure 15 – I ² C Register Write Transaction Functional Timing	33
Figure 16 – I ² C EEPROM Write Transaction Functional Timing	33
Figure 17 – I ² C EEPROM Read Status Transaction Functional Timing.....	33
Figure 18 – Interrupt Structure.....	34
Figure 19 - Electrical Characteristics: Clock Inputs	86
Figure 20 - Example External Components for Differential Input Signals	87
Figure 21 - Electrical Characteristics: CML Clock Outputs	88
Figure 22 – Example External Components for CML Output Signals	88
Figure 23 – Example External Components for HCSL Output Signals	89
Figure 24 - SPI Interface Timing.....	92
Figure 25 - I ² C Slave Interface Timing.....	93
Figure 26 - Non-customized Device Top Mark	94

List of Tables

Table 1 - Pin Descriptions.....	8
Table 2 - Crystal Selection Parameters	14
Table 3 - Default Input Clock Priorities	17
Table 4 – SPI Commands.....	30
Table 5 - Register Map	37
Table 6 - Recommended DC Operating Conditions	84
Table 7 - Electrical Characteristics: Supply Currents	84
Table 8 - Electrical Characteristics: Non-clock CMOS Pins	85
Table 9 - Electrical Characteristics: XA Clock Input	85
Table 10 - Electrical Characteristics: Clock Inputs, ICxP/N.....	86
Table 11 - Electrical Characteristics: CML Clock Outputs	87
Table 12 - Electrical Characteristics: CMOS and HSTL (Class I) Clock Outputs.....	89
Table 13 - Electrical Characteristics: APLL Frequencies	89
Table 14 - Electrical Characteristics: Jitter Specifications	90
Table 15 - Electrical Characteristics: Typical Output Jitter Performance, Configuration 2.....	90
Table 16 - Electrical Characteristics: Typical Output Jitter Performance, Configuration 3	90
Table 17 - Electrical Characteristics: Typical Input-to-Output Clock Delay	91
Table 18 - Electrical Characteristics: Typical Output-to-Output Clock Delay	91
Table 19 - Electrical Characteristics: SPI Slave Interface Timing, Device Registers.....	91
Table 20 - Electrical Characteristics: SPI Slave Interface Timing, Internal EEPROM	92
Table 21 - Electrical Characteristics: I ² C Slave Interface Timing	93
Table 22 – Package Top Mark Legend	94
Table 23 - 5x10mm LGA Package Thermal Properties.....	95

1. Application Examples

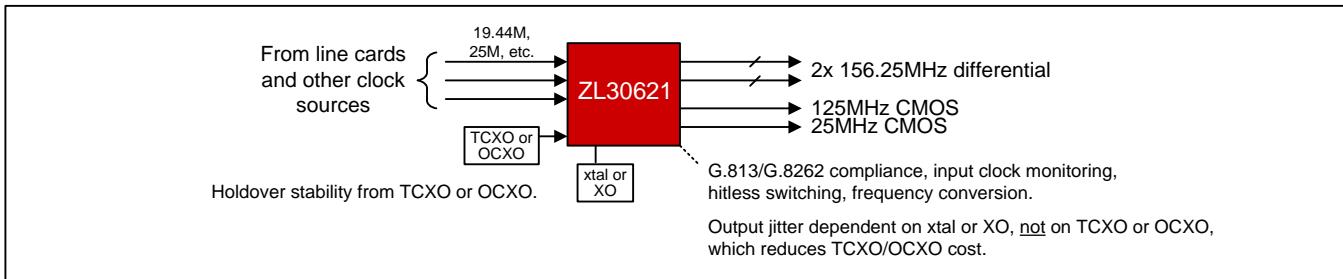


Figure 2 – Telecom Timing Card Application, TCXO + Crystal/XO for Lowest Jitter

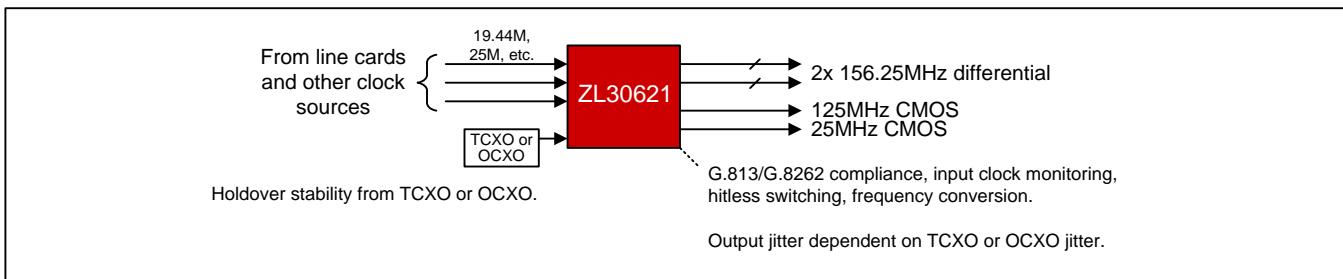


Figure 3 – Telecom Timing Card Application, TCXO Only

2. Detailed Features

2.1 Master Clock Jitter Attenuator and Multiplier

- Enables the DPLL to operate from any TCXO or OCXO $\geq 10\text{MHz}$ regardless of jitter
- When a low-cost crystal or XO is used, output jitter depends on crystal/XO, not on TCXO/OCXO jitter
- Reduces cost by removing tight jitter requirement from TCXO or OCXO

2.2 Input Block Features

- Up to three input clocks, two differential or single-ended, one single-ended
- Input clocks can be any frequency from 8kHz up to 1250MHz (differential) or 300MHz (single-ended)
- Supported telecom frequencies include PDH, SDH, Synchronous Ethernet, OTN, wireless
- Inputs constantly monitored by programmable activity monitors and frequency monitors
- Fast activity monitor can disqualify the selected reference after a few missing clock cycles
- Frequency measurement and monitoring with 1ppm resolution and accept/reject hysteresis
- Optional input clock invalidation on GPIO assertion to react to LOS signals from PHYs

2.3 DPLL Features

- Very high-resolution DPLL architecture
- State machine automatically transitions between tracking and freerun/holdover states
- Revertive or nonrevertive reference selection algorithm
- Programmable bandwidth from 0.1Hz to 10Hz
- Less than 0.1dB gain peaking
- Programmable phase-slope limiting
- Programmable tracking range (i.e. hold-in range)
- Truly hitless reference switching with <200ps output clock phase transient
- Output phase adjustment in 10ps steps
- High-resolution frequency and phase measurement
- Fast detection of input clock failure and transition to holdover mode
- Holdover frequency averaging with programmable averaging time and delay time

2.4 APLL Features

- Very high-resolution fractional scaling (i.e. non-integer multiplication)
- Any-to-any frequency conversion with 0ppm error
- Two high-speed dividers (integers 4 to 15, half divides 4.5 to 7.5)
- Easy-to-configure, completely encapsulated design requires no external VCXO or loop filter components

2.5 Output Clock Features

- Three low-jitter output clocks
- Each output can be one differential output or two CMOS outputs
- Output clocks can be any frequency from 1Hz to 1035MHz (250MHz max for CMOS and HSTL outputs)
- Output jitter as low as 0.25ps RMS (12kHz to 20MHz)
- In CMOS mode, an additional divider allows the OCxN pin to be an integer divisor of the OCxP pin (Example 1: OC3P 125MHz, OC3N 25MHz. Example 2: OC2P 25MHz, OC2N 1Hz)
- Outputs easily interface with CML, LVDS, LVPECL, HSTL, SSTL, HCSL and CMOS components
- Supported telecom frequencies include PDH, SDH, Synchronous Ethernet, OTN
- Sophisticated output-to-output phase alignment
- Per-output phase adjustment with high resolution and unlimited range
- Per-output enable/disable
- Per-output glitchless start/stop (stop high or low)

2.6 General Features

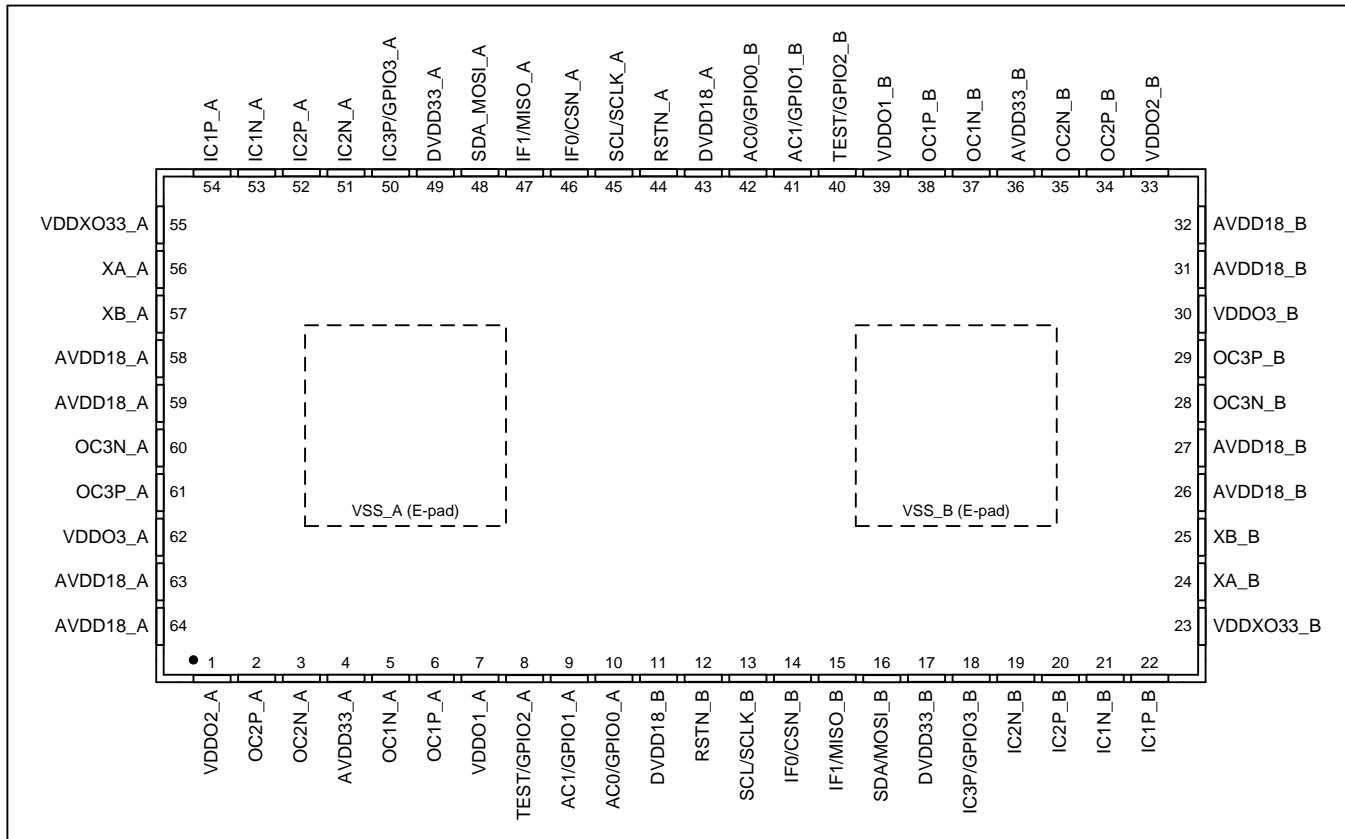
- SPI or I²C serial microprocessor interface
- Automatic self-configuration at power-up from internal EEPROM memory; pin control to specify one of four stored configurations
- Numerically controlled oscillator (NCO) mode allows system software to steer DPLL frequency with resolution better than 0.01ppb
- Input-to-output alignment with external feedback
- Up to eight general-purpose I/O pins each with many possible status and control options
- Output frame sync signals: 2kHz or 8kHz (SONET/SDH), 1Hz (IEEE 1588) or other frequency
- Internal compensation for local oscillator frequency error

2.7 Evaluation Software

- Simple, intuitive Windows-based graphical user interface
- Supports all device features and register fields
- Makes lab evaluation of the ZL30621 quick and easy
- Generates configuration scripts to be stored in internal EEPROM
- Generates full or partial configuration scripts to be run on a system processor
- Works with or without a ZL30621 evaluation board

3. Pin Diagram

The device is packaged in a 5x10mm 64-pin LGA.



4. Pin Descriptions

Block A pins names have “_A” suffix. Block B pin names have “_B” suffix. All inputs and outputs are LVCMOS unless described otherwise. The Type column uses the following symbols: I – input, I_{PU} – input with $50\text{k}\Omega$ internal pullup resistor, O – output, A – analog, P – power supply pin. All GPIO and SPI/I²C interface pins have Schmitt-trigger inputs and have output drivers that can be disabled (high impedance).

Table 1 - Pin Descriptions

Pin #	Name	Type	Description
54 53 52 51 50	IC1P_A IC1N_A IC2P_A IC2N_A IC3P/GPIO3_A	I I I I I/O	<p>Input Clock Pins Differential or Single-ended signal format. Programmable frequency.</p> <p><i>Differential:</i> See Table 10 for electrical specifications, and see Figure 20 for recommended external circuitry for interfacing these differential inputs to LVDS, LVPECL, CML or HSCL output pins on neighboring devices.</p>
22 21 20 19 18	IC1P_B IC1N_B IC2P_B IC2N_B IC3P/GPIO3_B	I I I I I/O	<p><i>Single-ended:</i> For input signal amplitude $>2.5\text{V}$, connect the signal directly to ICxP pin. For input signal amplitude $\leq 2.5\text{V}$, AC-coupling the signal to ICxP is recommended. Connect the N pin to a capacitor ($0.1\mu\text{F}$ or $0.01\mu\text{F}$) to VSS. As shown in Figure 20, the ICxP and ICxN pins are internally biased to approximately 1.3V. Treat the ICxN pin as a sensitive node; minimize stubs; do not connect to anything else including other ICxN pins.</p> <p><i>Unused:</i> Set ICEN.ICxEN=0. The ICxP and ICxN pins can be left floating.</p> <p>Note that the IC3N pin is not bonded out. A differential signal can be connected to IC3P by AC-coupling the POS trace to IC3P and terminating the signal on the driver side of the coupling cap. If not needed as an input clock pin, IC3P can behave as general-purpose I/O pin GPIO3, which is configured by GPIOCR2. Its state is indicated in GPIOSR.</p>
56 57 24 25	XA_A XB_A XA_B XB_B	A / I	<p>Crystal or Master Clock Pins <i>Crystal:</i> MCR1.XAB=01. An on-chip crystal driver circuit is designed to work with an external crystal connected to the XA and XB pins. See section 5.5.2 for crystal characteristics and recommended external components.</p> <p><i>Master Clock:</i> MCR1.XAB=10. An external local oscillator or clock signal (98-130MHz) can be connected to the XA pin. The XB pin must be left unconnected.</p>
6 5 2 3 61 60 38 37 34 35 29 28	OC1P_A OC1N_A OC2P_A OC2N_A OC3P_A OC3N_A OC1P_B OC1N_B OC2P_B OC2N_B OC3P_B OC3N_B	O	<p>Output Clock Pins CML, HSTL or 1 or 2 CMOS. Programmable frequency and drive strength. See Table 11 and Figure 22 for electrical specifications and recommended external circuitry for interfacing to LVDS, LVPECL or CML input pins on neighboring devices.</p> <p>See Table 12 for electrical specifications for interfacing to CMOS and HSTL inputs on neighboring devices.</p> <p>See Figure 23 for recommended external circuitry for interfacing to HCSL inputs on neighboring devices.</p>
44 12	RSTN_A RSTN_B	I_{PU}	<p>Reset (Active Low). When this global asynchronous reset is pulled low, all of the block's internal circuitry is reset to default values. The block is held in reset as long as RSTN is low. Microsemi recommends that RSTN_A and RSTN_B be wired together. See section 5.13.</p>

Table 1 - Pin Descriptions (continued)

Pin #	Name	Type	Description
10 9 42 41	AC0/GPIO0_A AC1/GPIO1_A AC0/GPIO0_B AC1/GPIO1_B	I/O	<p>Auto-Configure [1:0] / General Purpose I/O 0 and 1</p> <p><i>Auto Configure:</i> On the rising edge of RSTN these pins behave as AC[1:0] and specify one of the configurations stored in EEPROM. See section 5.4. Microsemi recommends AC0_A and AC0_B have the same value and AC1_A and AC1_B have the same value at reset.</p> <p><i>General-Purpose I/O:</i> After reset these pins are GPIO0 and GPIO1. GPIOCR1 configures the pins. Their states are indicated in GPIOSR.</p>
8 40	TEST/GPIO2_A TEST/GPIO2_B	I/O	<p>Factory Test / General Purpose I/O 2</p> <p><i>Factory Test:</i> On the rising edge of RSTN the pin behaves as TEST. Factory test mode is enabled when TEST is high. For normal operation TEST must be low on the rising edge of RSTN. Microsemi recommends TEST_A and TEST_B have the same value at reset.</p> <p><i>General-Purpose I/O:</i> After reset this pin is GPIO2. GPIOCR2 configures the pin. Its state is indicated in GPIOSR.</p>
46 14	IF0/CSN_A IF0/CSN_B	I/O	<p>Interface Mode 0 / SPI Chip Select (Active Low)</p> <p><i>Interface Mode:</i> On the rising edge of RSTN the pin behaves as IF0 and, together with IF1, specifies the interface mode for the block. See section 5.4.</p> <p><i>SPI Chip Select:</i> After reset this pin is CSN. When the block is configured as a SPI slave, an external SPI master must assert (low) CSN to access block registers.</p>
45 13	SCL/SCLK_A SCL/SCLK_B	I/O	<p>I²C Clock / SPI Clock</p> <p><i>I²C Clock:</i> When the block is configured as an I²C slave, an external I²C master must provide the I²C clock signal on the SCL pin. Note that I²C requires an external pullup resistor on this signal. See the I²C specification for details.</p> <p><i>SPI Clock:</i> When the block is configured as a SPI slave, an external SPI master must provide the SPI clock signal on SCLK.</p>
47 15	IF1/MISO_A IF1/MISO_B	I/O	<p>Interface Mode 1 / SPI Master-In-Slave-Out</p> <p><i>Interface Mode:</i> On the rising edge of RSTN the pin behaves as IF1 and, together with IF0, specifies the interface mode for the block. See section 5.4.</p> <p><i>SPI MISO:</i> After reset this pin is MISO. When the block is configured as a SPI slave, the block outputs data to an external SPI master on MISO during SPI read transactions.</p>
48 16	SDA/MOSI_A SDA/MOSI_B	I/O	<p>I²C Data / SPI Master-Out-Slave-In</p> <p><i>I²C Data:</i> When the block is configured as an I²C slave, SDA is the bidirectional data line between the block and an external I²C master. Note that I²C requires an external pullup resistor on this signal. See the I²C specification for details.</p> <p><i>SPI MOSI:</i> When the block is configured as a SPI slave, an external SPI master sends commands, addresses and data to the block on MOSI.</p>

Table 1 - Pin Descriptions (continued)

Pin #	Name	Type	Description
58, 59, 63, 64 26, 27, 31, 32	AVDD18_A	P	Analog Power Supply. 1.8V \pm 5%.
	AVDD18_B		
4 36	AVDD33_A AVDD33_B	P	Analog Power Supply. 3.3V \pm 5%.
43 11	DVDD18_A DVDD18_B	P	Digital Power Supply. 1.8V \pm 5%.
49 17	DVDD33_A DVDD33_B	P	Digital Power Supply. 3.3V \pm 5%.
7	VDDO1_A	P	Output OC1_A Power Supply. 1.5V to 3.3V \pm 5%.
1	VDDO2_A	P	Output OC2_A Power Supply. 1.5V to 3.3V \pm 5%.
62	VDDO3_A	P	Output OC3_A Power Supply. 1.5V to 3.3V \pm 5%.
39	VDDO1_B	P	Output OC1_B Power Supply. 1.5V to 3.3V \pm 5%.
33	VDDO2_B	P	Output OC2_B Power Supply. 1.5V to 3.3V \pm 5%.
30	VDDO3_B	P	Output OC3_B Power Supply. 1.5V to 3.3V \pm 5%.
55	VDDXO33_A	P	Analog Power Supply for Block A Crystal Driver Circuitry. 3.3V \pm 5%.
23	VDDXO33_B	P	Analog Power Supply for Block B Crystal Driver Circuitry. 3.3V \pm 5%.
E-pad	VSS_A	P	Ground. 0 Volts. For Block A.
E-pad	VSS_B	P	Ground. 0 Volts. For Block B.

5. Functional Description

5.1 Overview: Two Macro-Blocks

The device is composed of two independent macro-blocks, A and B. Block A has the G.8262-compliant DPLL while Block B has a simple jitter-attenuator DPLL. Both blocks have a low-jitter fractional-N APLL. Each block has its own independent set of pins: input clocks, output clocks, SPI/I²C interface, GPIO, reset and power supply pins. The two blocks have identical pin lists except Block A pin names end with “_A” while Block B pins names end with “_B”. Most of this data sheet uses pin names without these “_A” and “_B” endings, for example mentioning the TEST/GPIO2 pin rather than Block A’s TEST/GPIO2_A pin or Block B’s TEST/GPIO2_B pin. In this document any mention of a pin name without a block suffix should be understood to apply to that pin in both blocks. Block A’s registers are only accessible through Block A’s SPI/I²C interface, while Block B’s registers are only accessible through Block B’s SPI/I²C interface. Each block’s register map starts at address 0.

5.2 Device Identification

The 12-bit read-only ID field and the 4-bit revision field are found in each block’s **ID1** and **ID2** registers. Contact the factory to interpret the revision value and determine the latest revision.

5.3 Top-Level Configuration

5.3.1 Configurations 1 & 2: Block B as Master Clock Multiplier and/or Jitter Attenuator

Configurations 1 and 2 are shown in [Figure 5](#). In both configurations Block A is the main G.8262-compliant central timing function while Block B makes the 93MHz to 130MHz master clock frequency required by Block A’s DPLL and input block.

In configuration 1, Block B is configured as an APLL-only master clock multiplier, and its XA_B and XB_B pins are left unconnected. In this configuration Block A’s output jitter is dependent on TCXO jitter and TCXO frequency. All else being equal, lower TCXO frequency gives higher Block A output jitter in this configuration. For lowest possible

Block A output jitter, the ideal TCXO would have output frequency of 48MHz or higher and have an output jitter spec as close to 100fs (measured 12kHz to 5MHz) as possible.

In configuration 2, Block B is configured as a master clock multiplier and jitter attenuator and a crystal is connected to Block B's XA_B and XB_B pins. In this configuration output jitter is *independent* of TCXO jitter and TCXO frequency, and almost any TCXO can be used. Only an inexpensive 48-65MHz fundamental-mode crystal is required (see section 5.5.2 for crystal details). For applications where the device must have output jitter below 0.4ps the crystal should be replaced with an inexpensive 93-130MHz XO with a jitter spec of 100fs max (measured 12kHz to 5MHz). These XOs are available from several vendors. See section 5.5.1 for XO details. Device output jitter is typically 100-120fs lower when using an XO with Block B vs. using a crystal with Block B.

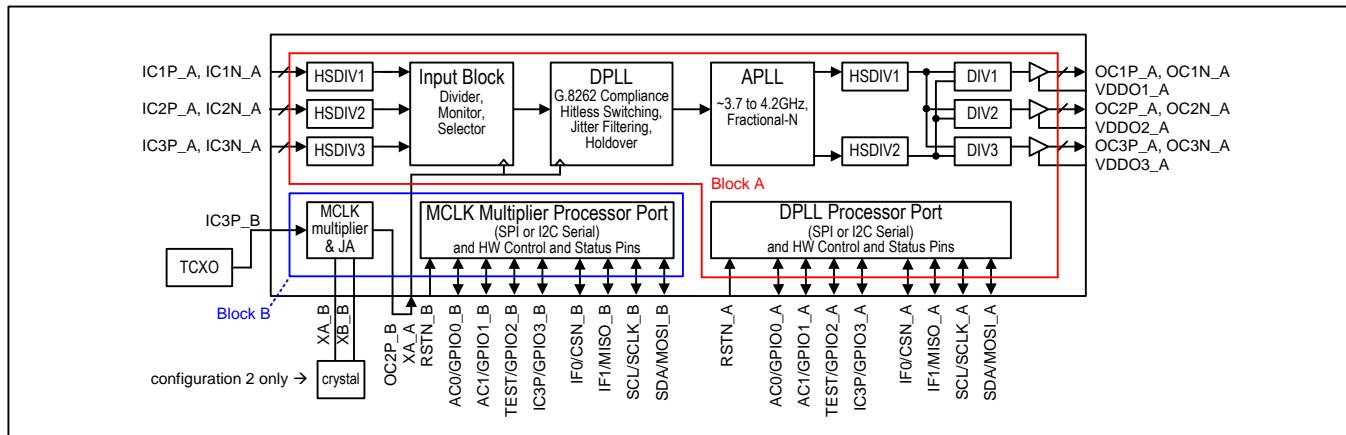


Figure 5 – Top-Level Configurations 1 and 2

5.3.2 Configuration 3: Block B as Output Frequency Converter and Jitter Attenuator

Configuration 3, shown in Figure 6, uses blocks A and B in an unconventional way to achieve lower >12kHz phase noise at the expense of higher <12kHz phase noise when compared to configuration 2. For applications where jitter requirements are only in the >12kHz band, configuration 3 is the lowest-jitter configuration.

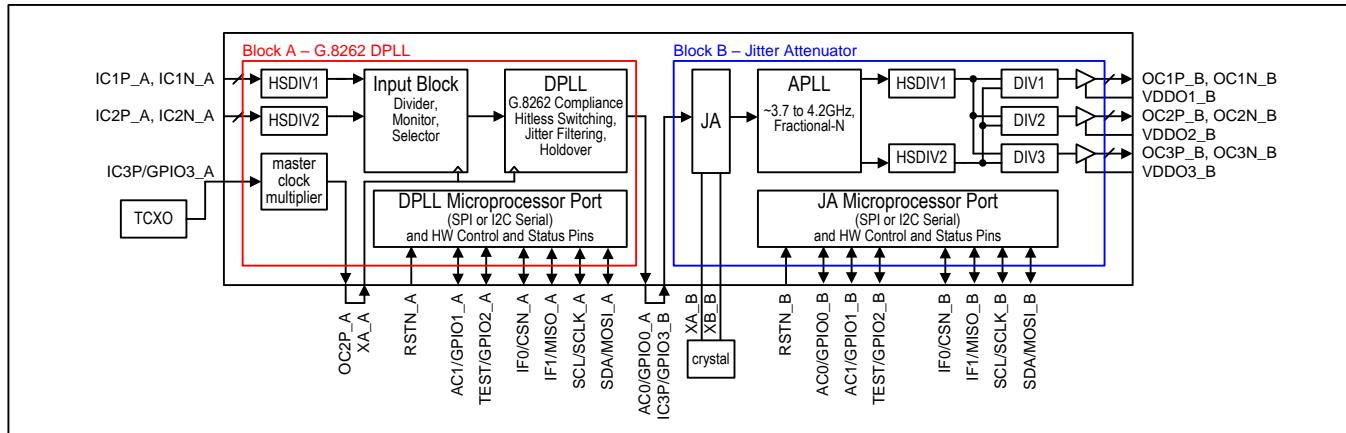


Figure 6 – Top-Level Configuration 3

In configuration 3, Block A's APLL is used to multiply the TCXO clock frequency to the 93MHz to 130MHz master clock frequency required by Block A's DLL and input block. In this configuration a jittery output from Block A's DLL is connected to Block B, which performs output frequency conversion and jitter attenuation. All input reference clocks are connected to Block A's inputs, and all synchronized output clocks come from Block B's outputs. In this configuration output jitter is *independent* of TCXO jitter and TCXO frequency, and almost any TCXO can be used. Only an inexpensive 48-65MHz fundamental-mode crystal is required (see section 5.5.2 for crystal details). For applications where the device must have output jitter below 0.4ps the crystal should be replaced with an inexpensive 93-130MHz XO with a jitter spec of 100fs max (measured 12kHz to 5MHz). These XOs are

available from several vendors. See section [5.5.1](#) for XO details. Device output jitter is typically 100-120fs lower when using an XO with Block B vs. using a crystal with Block B.

If device output jitter must be less than 1ps then the DPLL's phase detector frequency should be $\geq 64\text{kHz}$. This means that all input clock frequencies should be $\geq 64\text{kHz}$. It also means that all input clock frequencies should have a common divisor $\geq 64\text{kHz}$.

Before selecting configuration 3 it is very important to understand that configuration 3's phase noise less than 12kHz (offset from the carrier) is significant and varies with input clock frequency and frequency offset. If the output jitter of the ZL30621 is measured using a band-pass filter with 20dB/decade rolloff at the 12kHz (or similar low-end) corner then configuration 3's phase noise less than 12kHz will cause higher and more variable output jitter. When jitter must be measured this way then configuration 2 is generally the better choice. If, however, ZL30621 output jitter is measured using "brick wall" filtering (i.e. infinite-slope rolloff at the 12kHz corner, which is the filter that the widely-used Agilent E5052B phase noise analyzer uses) then configuration 3 is the lowest-jitter choice and its output jitter is consistently low (i.e. it does not vary with input clock frequency and offset).

5.3.3 Configuration Comparison

The following table compares configurations 1 through 3.

	Configuration 1	Configuration 2	Configuration 3
Jitter independent of TCXO jitter	No	Yes	Yes
Jitter independent of TCXO frequency	No	Yes	Yes
Crystal required in addition to TCXO	No	Yes	Yes
More-expensive low-jitter, higher-frequency TCXO required for low-jitter applications	Yes	No	No
Configuration 2 vs. configuration 3: $>12\text{kHz}$ phase noise comparison		higher	lower
Configuration 2 vs. configuration 3: $<12\text{kHz}$ phase noise comparison		lower	higher
Number of reference clock inputs	3	3	2

In both configuration 2 and configuration 3 Block B is configured as jitter attenuator and uses its own simple DPLL to accomplish jitter attenuation. Block B DPLL bandwidth is configurable between 20Hz and 100Hz to accommodate systems with different thermal transients: higher bandwidth for bigger thermal transients (temperature control by turning system fans on and off) and lower bandwidth for smaller thermal transients (no system fans).

5.3.4 Evaluation Software for Device Configuration

Microsemi provides evaluation software that gives the user a simple, intuitive graphical user interface in which to generate complete device configurations. Often customers can generate base device configurations with the evaluation software without learning the device register set in detail. This saves time and money during system development. Use of the evaluation software is required as explained in sections [5.8.1](#) and [5.7.2.3](#).

5.4 Pin-Controlled Automatic Configuration at Reset

The device configuration is determined at reset (i.e. on the rising edge of RSTN) by the signal levels on five pins on each macro-block: TEST/GPIO2, AC1/GPIO1, AC0/GPIO0, IF1/MISO and IF0/CSN. For these pins, the first name (TEST, AC1, AC0, IF1, IF0) indicates their function when they are sampled by the rising edge of the RSTN pin. The second name refers to their function after reset. The values of these pins are latched into the [CFGSR](#) register when RSTN goes high. The reset values of the Block A pins must match the reset value of the Block B pins for proper operation. To ensure the device properly samples the reset values of these pins, the following guidelines should be followed:

1. Any pullup or pulldown resistors used to set the value of these pins at reset should be $1\text{k}\Omega$.
2. RSTN must be asserted at least as long as specified in section [5.13](#).

The hardware configuration pins are grouped into three sets:

1. TEST - Manufacturing test mode
2. IF[1:0] – Microprocessor interface mode and I²C address
3. AC[1:0] – Auto-configuration from EEPROM

The TEST pin selects manufacturing test modes when TEST=1 (the AC[1:0] pins specify the test mode). TEST=1 and AC[1:0]=00 configures the device so that production SPI EEPROM programmers can program the internal EEPROM. See section [5.15.2](#) for more information.

The IF[1:0] pins specify the processor interface mode and the I²C addresses:

IF1	IF0	Processor Interface
0	0	I ² C, Block A slave address 10000 00 Block B slave address 10001 00
0	1	I ² C, Block A slave address 10000 01 Block B slave address 10001 01
1	0	I ² C, Block A slave address 10000 10 Block B slave address 10001 10
1	1	SPI Slave

The AC[1:0] pins specify which of four configurations in the EEPROM to execute after reset:

AC1	AC0	Auto Configuration
0	0	Configuration 0
0	1	Configuration 1
1	0	Configuration 2
1	1	Configuration 3

For more information about auto-configuration from EEPROM see section [5.15](#).

5.5 Local Oscillator or Crystal

In all three configurations described in section [5.3](#), a G.8262-quality TCXO is required to be connected to an IC3P pin. In addition to this TCXO, configurations 2 and 3 call for an external local oscillator or an external crystal to be connected to the XA_B pin. Section [5.5.1](#) describes how to connect an external oscillator to a macro-block and the required characteristics of the oscillator. Section [5.5.2](#) describes how to connect an external crystal to the on-chip crystal driver circuit and the required characteristics of the crystal.

5.5.1 External Oscillator

A signal from an external oscillator can be connected to the XA_B pin (XB_B must be left unconnected). [Table 9](#) specifies the range of possible frequencies for the XA_B input. Several vendors including Vectron, Rakon and TXC offer low-cost, low-jitter XOs with output frequencies in this range. The frequency of the external oscillator must be specified Block B's [MCR2.MCLK](#) field. To minimize jitter, the signal must be properly terminated and must have very short trace length. A poorly terminated single-ended signal can greatly increase output jitter, and long single-ended trace lengths are more susceptible to noise. When [MCR1.XAB=10](#) in Block B, XA is enabled as a single-ended input.

5.5.2 External Crystal and On-Chip Driver Circuit

Block B's on-chip crystal driver circuit is designed to work with a fundamental mode, AT-cut crystal resonator. See [Table 2](#) for recommended crystal specifications. To enable the crystal driver, set [MCR1.XAB=01](#).

See [Figure 7](#) for the crystal equivalent circuit and the recommended external capacitor connections. To achieve a crystal load (C_L) of 10pF, an external 16pF is placed in parallel with the 4pF internal capacitance of the XA pin, and an external 16pF is placed in parallel with the 4pF internal capacitance of the XB pin. The crystal then sees a load of 20pF in series with 20pF, which is 10pF total load. Note that the 16pF capacitance values in [Figure 7](#) include all capacitance on those nodes. If, for example, PCB trace capacitance between crystal pin and IC pin is 2pF then 14pF capacitors should be used to make 16pF total.

The crystal, traces, and two external capacitors should be placed on the board as close as possible to the XA and XB pins to reduce crosstalk of active signals into the oscillator. Also no active signals should be routed under the crystal circuitry.

Note: Crystals have temperature sensitivities that can cause frequency changes in response to ambient temperature changes. In applications where significant temperature changes are expected near the crystal, it is recommended that the crystal be covered with a thermal cap, or an external XO or TCXO should be used instead.

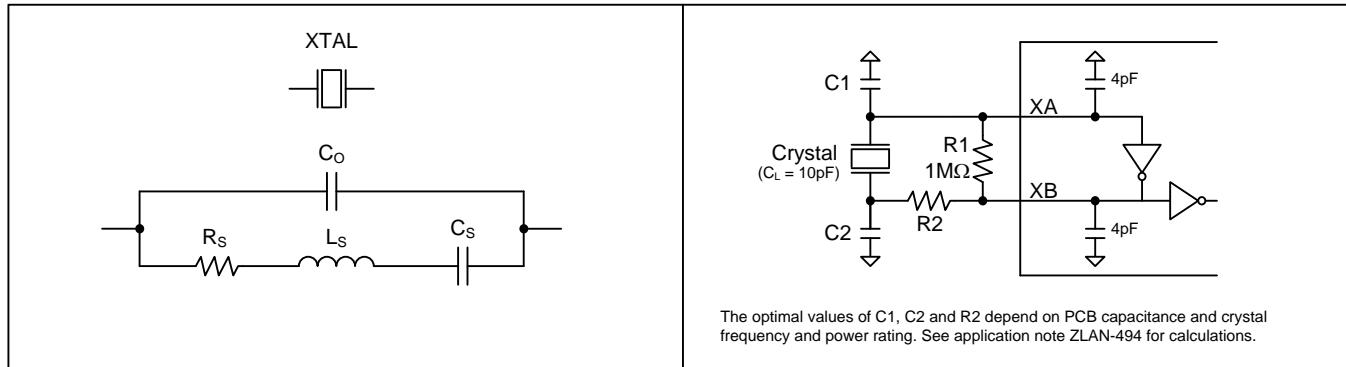


Figure 7 - Crystal Equivalent Circuit / Recommended Crystal Circuit

Table 2 - Crystal Selection Parameters

Parameter	Symbol	Min.	Typ.	Max.	Units
Crystal oscillation frequency ¹	fosc	25		60	MHz
Shunt capacitance	C ₀		2	5	pF
Load capacitance	C _L		10		pF
Equivalent series resistance (ESR) ²	R _S			60	Ω
	R _S			50	Ω
Maximum crystal drive level		100			μW

Note 1: Higher frequencies give lower output jitter, all else being equal.

Note 2: These ESR limits are chosen to constrain crystal drive level to less than 100μW. If the crystal can tolerate a drive level greater than 100μW then proportionally higher ESR is acceptable.

Parameter	Symbol	Min.	Typ.	Max.	Units
Crystal Frequency Stability vs. Power Supply	f _{FVD}		0.2	0.5	ppm per 10% Δ in VDD

5.5.3 Ring Oscillator (for System Start-Up)

To ensure that registers can be written immediately after system start-up, in its power-on reset state each macro-block operates its registers and processor interface from an internal ring oscillator.

For Block B when configured for jitter attenuation and always for Block A, as soon as the external oscillator connected to the block's XA pin has stabilized and is ready to use, the [MCR1.MCSEL](#) bit must be set to source the DPLL master clock from XA. If the ring oscillator causes undesirable spurs it can be disabled (powered down) by setting [MCR1.ROSCD=1](#).

5.6 Input Signal Format Configuration

Each macro-block's input clocks IC1, IC2 and IC3 are enabled by setting the enable bits in the [ICEN](#) register. The power consumed by a differential receiver is shown in [Table 7](#). The electrical specifications for these inputs are listed in [Table 10](#). Each input clock can be configured to accept nearly any differential signal format by using the proper set of external components (see [Figure 20](#)). To configure these differential inputs to accept single-ended CMOS signals, connect the single-ended signal to the ICxP pin, and connect the ICxN pin to a capacitor (0.1μF or 0.01μF) to VSS. Each ICxP and ICxN pin is internally biased to approximately 1.3V. If an input is not used, both

ICxP and ICxN pins can be left floating. Note that the IC3N pin is not present on either macro-block. A differential signal can be connected to IC3P by AC-coupling the POS trace to IC3P and terminating the signal on the driver side of the coupling cap. If not needed as an input clock pin, IC3P can behave as general-purpose I/O pin GPIO3.

5.7 Input Block: Input Divider, Monitor and Selector

All of the functions and registers mentioned in this section are in Block A.

The input block performs the following functions:

- Frequency division to a frequency suitable for DPLL locking
- Activity monitoring
- Frequency monitoring
- DPLL input clock selection (automatic or manual)

Figure 8 is a block diagram of the input block. This block requires a master clock as described in section 5.3.

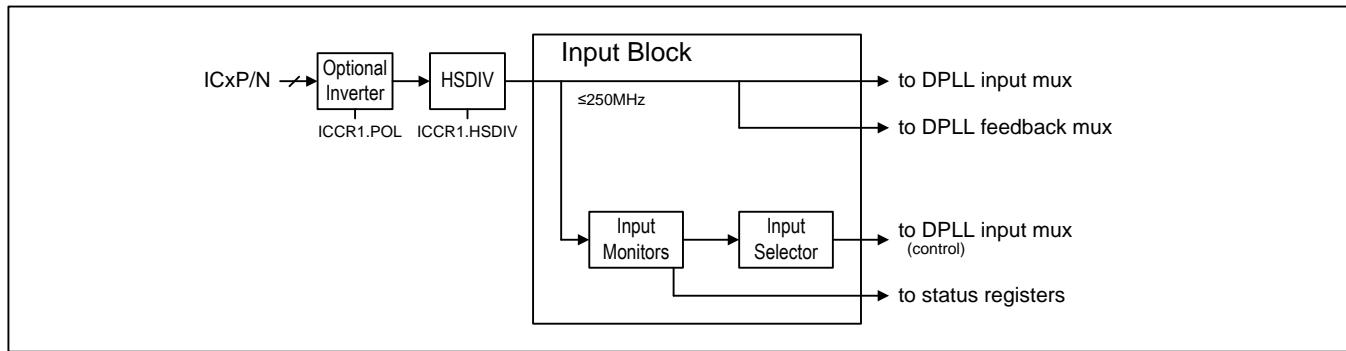


Figure 8 - Input Block Diagram

5.7.1 Input Clock Inversion and High-Speed Dividers

Block A's input block tolerates a wide range of duty cycles out to a minimum high time or minimum low time of 3ns or 30% of the clock period, whichever is smaller.

Any frequency in the 8kHz to 250MHz range can be accepted by the input block. Important notes about the input block:

- **ICxCR1.POL** specifies the edge to which the DPLL will lock (by default, the rising edge).
- **ICxCR1.HSDIV** must be set correctly to reduce the clock frequency below 250MHz.
- The frequencies of all enabled Block A ICx input clocks must divide by integers to a common DPLL phase-frequency detector (PFD) frequency $\geq 8\text{kHz}$. In addition, the common PFD frequency must be ≥ 20 times the DPLL bandwidth.

5.7.2 Input Clock Monitoring

Each Block A ICx input clock is continuously monitored for activity and frequency accuracy.

The activity monitor counts the number of input clock cycles that occur during a configurable interval. This provides the fastest detection when the input clock is stopped or far off frequency. Register bit **ICxSR.ACVAL** indicates the real-time status of this monitor. The ACVAL bit stays low when the input clock is not toggling or its frequency is grossly too high; ACVAL flickers (i.e. rapidly changes states) when the input clock is toggling but its frequency is grossly too low.

Frequency monitoring is handled by either a percent frequency monitor (1% to 20% in 1% steps) or a ppm frequency monitor (1ppm to 500ppm in 1ppm steps) or a combination of both depending on the level of accuracy

required. As with any frequency measurement, there is a tradeoff between accuracy and measurement time. Higher accuracy ppm monitoring requires longer measurement time (from tens of milliseconds to seconds) while lower accuracy percent monitoring can complete a measurement in as little as 10 μ s. Register bits **ICxSR.PCVAL** and **ICxSR.PPVAL** indicate the real-time status of the percent and ppm monitors, respectively.

Any input clock that fails activity monitoring or frequency monitoring is declared invalid. The valid/invalid state of each input clock is reported in the corresponding real-time status bit in the **VALSR** registers. When the valid/invalid state of a clock changes, the corresponding latched status bit is set in the **VALSR** registers. Input clocks manually marked invalid in the DPLL's **VALCR1** register cannot be automatically selected as the reference for the DPLL.

The activity monitor, percent and ppm monitors can be enabled and disabled using the ACEN, PCEN and PPEN bits, respectively in the **MONxCR2** register.

In addition to the monitors in the input block, the DPLL can also invalidate an input. If the input is the DPLL's selected reference and the DPLL cannot lock within the time specified by the **PHLKTO** register, the DPLL invalidates the input by setting the **ICxSR.LKTO** bit.

Note 1: The ppm monitor reject threshold must be ≥ 3 ppm higher than the accept threshold.

5.7.2.1 External Monitoring

Some clock signals come from external components that can monitor the quality of a clock signal or the quality of a signal from which the clock signal is derived. One example is a BITS receiver in telecom equipment. This component receives a DS1, E1 or 2048kHz synchronization signal and recovers a clock from that signal. The BITS receiver monitors the incoming signal and can declare loss of signal (LOS), loss of frame alignment (LOF) and other defects in the incoming signal. Another example is a Synchronous Ethernet PHY, which receives an Ethernet signal and recovers a clock from that signal. This PHY can declare loss of lock, loss of codeword alignment and other defects.

When a neighboring component can detect that the incoming signal or the clock recovered from the signal is somehow out of specification, a loss-of-signal indication from that component can be connected to a Block A GPIO pin to instantly invalidate the input clock. Any of the Block A's unused GPIO pins can be used as a loss-of-signal indicator for any of the IC1, IC2 or IC3 input clocks. **ICxSR.LOS** indicates the real-time LOS status from the GPIO pin.

Example: Configure GPIO1 to be the active-low LOS signal for IC1:

GPIOCR1.GPIO1C=0001 (Configure GPIO1 to be an input with inversion)
MON1CR2.LOSSS=010 (Configure the IC1 monitor's LOS source to be GPIO1)

5.7.2.2 Monitor Priority and Validation Timer

All enabled input monitors must declare an input valid for a configurable duration (which can be zero) before the input clock is validated and considered eligible for selection as the DPLL's selected reference. The monitors have a priority hierarchy in which an invalid declaration by a higher-priority monitor forces an invalid declaration in all lower-priority monitors. When a valid higher-priority monitor declares the input valid, the next lower priority monitor can then initiate its validation process. The monitor hierarchy is as follows:

- Input LOS from a GPIO pin forces all other monitors (activity and frequency) to declare the input invalid.
- When the activity monitor declares invalid, it forces the frequency monitors to declare invalid.

When a monitor is not enabled, it continually declares the input clock valid.

After all monitors declare an input clock valid (**ICxSR.VAL=1**) the validation timer requires all the monitors to continue to indicate the clock is valid for a configurable validation time before the input is declared valid for use as a DPLL input (**ICxSR.VALT=1**).

5.7.2.3 Input Monitor Configuration

The device's input monitors are very sophisticated, but the configuration registers for these monitors are, generally speaking, low-level coefficients rather than user concepts such as ppm. As a result most input monitor registers are not documented in this data sheet. Instead, Microsemi provides evaluation software that gives the user a simple, intuitive graphical user interface in which to generate complete device configurations, including all aspects of input monitor behavior. Configuration files from the evaluation software can be stored in internal EEPROM to allow the device to self-configure at reset. Alternately, system software can perform the register writes listed in the configuration files as needed to configure/reconfigure the device.

5.7.3 Input Clock Priority, Selection and Switching for the DPLL

All of the functions and registers mentioned in this section are in Block A.

5.7.3.1 Priority Configuration

During normal operation, the selected reference for the DPLL is chosen automatically based on the priority rankings assigned to the input clocks in Block A's input priority registers ([IPR1](#) and [IPR2](#)). The default input clock priorities are shown in [Table 3](#).

Any unused input clock should be given the priority value 0, which disables the clock and marks it as unavailable for selection. Priority 1 is highest.

Table 3 - Default Input Clock Priorities

INPUT CLOCK	DPLL DEFAULT PRIORITY
IC1_A	1
IC2_A	2
IC3_A	3

5.7.3.2 Automatic Selection

When [ICSCR1.EXTSW=0](#), automatic input clock reference selection is used for the DPLL. The input reference selection algorithm chooses the highest-priority valid input clock to be the selected reference. The real-time valid/invalid state of each input clock is maintained in the [VALSR](#) registers (see section [5.7.2](#)). The priority of each input clock is set as described in section [5.7.3.1](#). To select the DPLL input clock based on these criteria, the selection algorithm maintains a priority table of valid inputs. The top entry in this priority table and the selected reference are indicated in the [PTAB1](#) register. The second- and third-priority inputs are indicated in the [PTAB2](#) register.

If two or more input clocks are given the same priority number then those inputs are prioritized among themselves using a fixed circular list. If one equal-priority clock is the selected reference but becomes invalid then the next equal-priority clock in the list becomes the selected reference. If an equal-priority clock that is not the selected reference becomes invalid, it is simply skipped over in the circular list. The selection among equal-priority inputs is inherently nonrevertive, and revertive switching mode (see next paragraph) has no effect in the case where multiple equal-priority inputs have the highest priority.

An important input to the selection algorithm is the REVERT bit in the [ICSCR1](#) register. In revertive mode (REVERT=1), if an input clock with a higher priority than the selected reference becomes valid, the higher priority reference immediately becomes the selected reference. In nonrevertive mode (REVERT=0), the higher priority reference does not immediately become the selected reference but does become the highest priority reference in the priority table ([PTAB1.REF1](#)). (The selection algorithm always switches to the highest-priority valid input when the selected reference goes invalid, regardless of the state of the REVERT bit.) For many applications, nonrevertive mode is preferred because it minimizes disturbances on the output clocks due to reference switching.

In nonrevertive mode, planned switchover to a newly-valid higher priority input clock can be done manually under software control. The validation of the new higher priority clock sets the corresponding latched status bit in the

[VALSR](#) registers, which can drive an interrupt request if needed. System software can then respond to this change of state by briefly enabling revertive mode (toggling REVERT high then back low) to force the switchover to the higher priority clock.

5.7.3.3 Manual Selection

The bits of the [VALCR1](#) register can be used to perform manual selection of an input clock. When all input clocks have non-zero priorities in the [IPR](#) registers, an input clock can be manually selected by setting the [VALCR1](#) bit for that input clock to 1 and the [VALCR1](#) bits for the other input clocks to 0.

5.7.3.4 External Reference Switching Mode

In this mode a GPIO pin controls reference switching between two input clocks. This mode is enabled by setting the [ICSCR1.EXTSW](#)=1. In this mode, if the GPIO pin is high, the DPLL is forced to lock to input IC1 (if the priority of IC1 is nonzero in [IPR1](#)) or IC3 (if the priority of IC1 is zero) whether or not the selected input has a valid reference signal. If the GPIO pin is low the DPLL is forced to lock to input IC2 whether or not IC2 has a valid reference signal. The GPIO pin is selected by [MCR2.EXTSS](#).

In external reference switching mode the input selector logic behaves as a simple 2:1 mux, and the DPLL is forced to try to lock to the selected reference whether it is valid or not. This mode controls the [PTAB1.SELREF](#) field directly and, therefore, is not affected by the state of the [ICSCR1.REVERT](#) bit. During external reference switching mode, only [PTAB1.SELREF](#) is affected; the [PTAB1.REF1](#) field continues to indicate the highest-priority valid input chosen by the automatic selection logic. The priorities of IC1, IC2 and IC3 in the [IPR](#) registers must be non-zero for proper behavior in external reference switching mode.

5.8 DPLL Architecture and Configuration

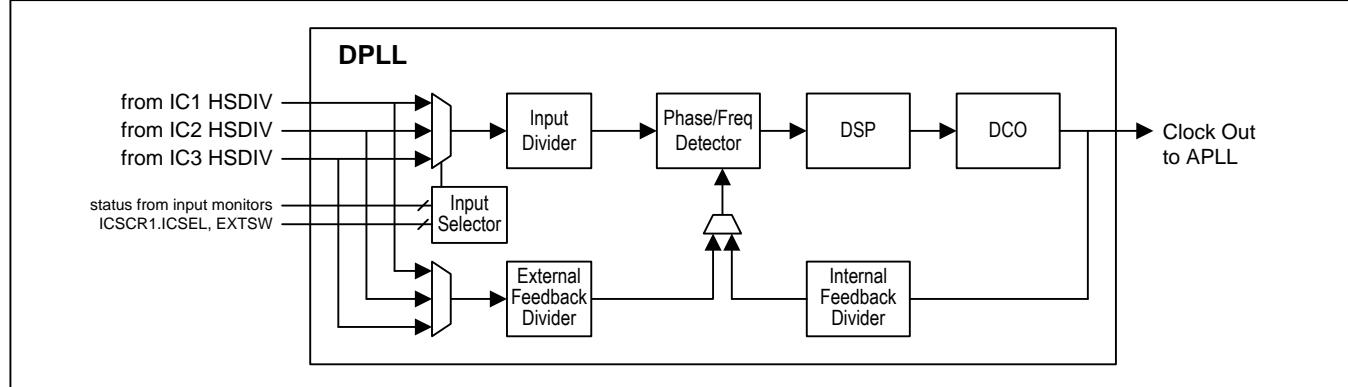


Figure 9 - DPLL Block Diagram

Digital PLLs have two key benefits: (1) stable, repeatable performance that is insensitive to process variations, temperature, and voltage; and (2) flexible behavior that is easily configured and reprogrammed. DPLLs use a digitally controlled oscillator (DCO) to generate the DPLL output clock. The DPLL output clock is then provided to an APLL for clock multiplication/frequency conversion.

The DPLL in the device is configurable for many PLL parameters including bandwidth, input frequency, pull-in/hold-in range, input-to-output phase offset, and more. No knowledge of loop equations or gain parameters is required to configure and operate the device. No external components are required for the DPLL except a local oscillator connected to the XA_A pin to provide the DPLL's master clock.

5.8.1 DPLL Configuration

The DPLL is very sophisticated, but the configuration registers for the DPLL are, generally speaking, very low-level coefficients rather than user concepts such as bandwidth and pull-in range. As a result most DPLL registers are not documented in this data sheet. Instead, Microsemi provides evaluation software that gives the user a simple,

intuitive graphical user interface in which to generate complete device configurations, including all aspects of DPLL behavior. Configuration files from the evaluation software can be stored in internal EEPROM to allow the device to self-configure at reset. Alternately, system software can perform the register writes listed in the configuration files as needed to configure/reconfigure the device. The most frequently used DPLL status register fields and real-time control register fields are documented in section 6.3.6 and discussed in the DPLL sections below.

Microsemi provides base configuration files for G.8262 option 1 and option 2 compliance. Customers should start by loading one of these files into the evaluation software and then further refine the device configuration with the desired I/O frequency plan and other application details.

5.8.2 DPLL States

Tracking (Locked and Unlocked). When a valid input clock is available, the DPLL is in the tracking state ([DSRR1.TRK=1](#)) and is either locked to an input clock ([DSRR1.LOL=0](#)) or unlocked ([DSRR1.LOL=1](#)). While locked to an input clock the DPLL can be configured to average its fractional frequency offset over time to calculate a holdover average. The DPLL uses a fast-lock behavior to decrease lock time. When the DPLL is tracking (TRK=1), fast-lock is indicated by [DSRR1.FLOCK=1](#).

Freerun/Holdover. When all input clocks become invalid, the DPLL enters the freerun/holdover state ([DSRR1.TRK=0](#)) in which it operates open-loop. In this mode the DPLL can be in freeun ([DSRR1.HO=0](#)) in which its output frequency has the same fractional frequency offset as the master clock signal. Or the DPLL can be in holdover ([DSRR1.HO=1](#)) in which the output frequency has the average fractional frequency offset calculated previously when the DPLL was locked to an input clock. The DPLL can automatically transition from the freerun/holdover state to the tracking state when an input clock is declared valid. The holdover average can be retained or cleared when the DPLL switches input clocks. The [DPLLCR1.HOMODE](#) field controls DPLL freerun/holdover behavior.

System software can manually force the DPLL into the freerun/holdover state as needed using the [DPLLCR1.MODE](#) field.

5.8.3 DPLL Capabilities

Bandwidth. The DPLL can be configured for any bandwidth from 0.1Hz to 10Hz. The DPLL can easily be configured for a damping factor that ensures gain peaking near the DPLL's corner frequency is minimized.

Pull-In/Hold-In Range. The DPLL tracking range is configurable from $\pm 1\text{ppm}$ to $\pm 1000\text{ppm}$. The DPLL reports when it has reached the limit of the range in the [DSRR2.FLIM](#) register bit. The DPLL's hold-in range is the same as its tracking range. The DPLL's pull-in range should be considered to be half the size of the tracking range for reasonable pull-in time. For example, when tracking range is $\pm 1000\text{ppm}$, pull-in range should be considered to be $\pm 500\text{ppm}$. Note that the ppm input monitor's accept threshold determines the pull-in range when the accept threshold is lower than the DPLL tracking range, and the monitor's reject threshold determines the hold-in range when the reject threshold is lower than the DPLL tracking range.

Programmable Lock Criteria. The DPLL has configurable criteria for defining when it declares lock. In addition to phase, the DPLL can also be configured to declare loss of lock when its fractional frequency offset exceeds the DPLL's tracking range.

Programmable Phase Lock Timeout. When the DPLL fails to lock to the selected input clock within the timeout duration specified by the [PHLKTO](#) register, the input is declared invalid by the input block, which sets the [ICxSR.LKTO](#) bit.

Frequency and Phase Reporting. The DPLL reports in real-time its frequency (i.e. fractional frequency offset in ppb/ppm vs. its nominal frequency) and its phase vs. the input clock signal. DPLL frequency resolution is better than 0.005ppb. DPLL frequency offset is reported in the [DFREQ](#) registers. DPLL phase is reported in the [DPHASE](#) registers.

Phase-Slope Limiting (PSL). In the tracking state the DPLL can limit its phase rate-of-change to ensure that consumers of the device's output clocks can follow the phase change. The DPLL can be configured for a wide

range of phase-slope limits. See the evaluation software for details. The DPLL reports when it is currently limiting its phase rate-of-change in the [DSRR2.PSLIM](#) register bit. Note that PSL can affect pull-in time and ability to pull in and must be used with care. Contact Microsemi for assistance. In general PSL in ns/sec should be $\geq 10x$ the DPLL bandwidth in Hz. Also, DPLL pull-in range in ppm is limited to the PSL value expressed in μ sec.

Absolute Phase Locking and Relative Phase Locking. The DPLL has two modes of tracking an input reference clock: absolute phase locking and relative phase locking. In absolute phase locking the DPLL's tracking goal is the absolute difference between input and feedback clocks specified by a DPLL parameter. Often this parameter (DPHOFF) is set to have the DPLL lock with zero phase difference. Using a non-zero value enables DPLL input-to-output phase adjustment.

In relative phase locking whenever a new input clock is selected, the DPLL measures the difference between input phase and feedback phase and sets that value as the tracking goal. Relative phase locking supports hitless switching. The difference between input phase and feedback phase is averaged for a programmable duration to determine the tracking goal.

Hitless Switching. The DPLL can perform truly hitless switching (max 200ps output phase change) between two input clocks that (1) have been divided by the input dividers to the same nominal PFD frequency, and (2) have the same fractional frequency offset (i.e. they are traceable to the same reference). The two input clocks can have any arbitrary phase relationship and can be any frequency in the PFD input frequency range (as long as the two are the same frequency). Note: when DPLL PFD input frequency is ≤ 16 kHz, output phase change can be up to 1ns with low-jitter/wander input clocks and could be somewhat higher with high-jitter/wander input clocks.

Holdover Averaging. The holdover averaging calculation has a programmable averaging window and a programmable throw-away window. The throw-away window specifies an interval of time, just prior to the DPLL entering holdover, that is not included in the average. When an input clock is invalidated by the input monitor for frequency error, the throw-away window allows the DPLL to exclude the worst of the frequency excursion from the holdover average.

Numerically Controlled Oscillator (NCO) Mode. With this behavior the DPLL is in holdover and system software controls the DPLL's output frequency using the 40-bit FREQZ field in the the [DFREQZ](#) registers. The resolution of frequency control is better than 0.01ppb. See section [5.8.7](#) for more details.

5.8.4 Input Wander and Jitter Tolerance

Wander is tolerated up to the point where wander causes an apparent long-term frequency offset larger than the frequency threshold set in the input monitor. In such a situation the input clock would be declared invalid. Jitter can be tolerated up to the point of eye closure. The high-jitter input clock signal should be divided down to a lower frequency by the DPLL's input divider for high jitter tolerance.

5.8.5 Jitter and Wander Transfer

The transfer of jitter and wander from the selected reference to the output clocks has a programmable transfer function that is determined by the DPLL bandwidth. The -3dB corner frequency of the jitter transfer function can be set to any value from 5Hz to 500Hz.

During locked mode, the transfer of wander from the local oscillator clock (connected to the XA pin) to the output clocks is not significant as long as the oscillator is stable enough and the DPLL bandwidth is set high enough to allow the DPLL to quickly compensate for oscillator frequency changes. During freerun/holdover, local oscillator wander has a much more significant effect. See section [5.5.1](#).

5.8.6 Output Jitter and Wander

Several factors contribute to jitter and wander on the output clocks, including:

- Jitter and wander amplitude on the selected reference (while in the locked state)
- The jitter/wander transfer characteristic of the device (while in the locked state)

- The jitter and wander on the local oscillator clock signal (especially wander while in the holdover state)

The DPLL has programmable bandwidth (see Section 5.8.3). With respect to jitter and wander, the DPLL behaves as a low-pass filter with a programmable pole. The bandwidth of the DPLL is normally set low enough to strongly attenuate jitter. The wander and jitter attenuation depends on the DPLL bandwidth chosen.

5.8.7 Numerically Controlled Oscillator (NCO) Mode

With this behavior the DPLL is in holdover and system software controls the DPLL's output frequency using the 40-bit FREQZ field in the the [DFREQZ](#) registers. The resolution of frequency control is better than 0.01ppb. The device can smoothly switch from locking to an input reference to NCO behavior and vice versa without phase hits.

The nominal FREQZ value, hereafter referred to as FREQZ0, is computed by the evaluation software for the desired device configuration. When the FREQZ field is set to the FREQZ0 value, the device's output clock frequencies have a fractional frequency offset of zero with respect to the NCO master clock signal applied to the XA pin.

(Fractional frequency offset (FFO) is defined as $(\text{actual_frequency} - \text{nominal_frequency}) / \text{nominal_frequency}$. FFO is a unitless number but is typically expressed in parts per billion (ppb), parts per million (ppm) or percent.)

To control the NCO, system software first reads the FREQZ0 value from the device. FREQZ0 is a 40-bit unsigned integer.

To change the NCO frequency to a specific FFO (in ppm), system software calculates newFREQZ (a 40-bit unsigned integer) as follows:

$$\text{newFREQZ} = \text{round}(\text{FREQZ0} * (1 + \text{FFO}/1\text{e}6))$$

System software then writes the newFREQZ value directly to the FREQZ field in the [DFREQZ](#) registers.

Note that any subsequent frequency changes are calculated using the same equation from the original FREQZ0 value and are not a function of the previous newFREQZ value. The value of newFREQZ should be kept within $\pm 1000\text{ppm}$ of FREQZ0 and within $\pm 500\text{ppm}$ of the previous newFREQZ value to avoid causing the APLL to lose lock.

5.9 APLL Configuration

5.9.1 APLL Signal Source

In top-level configurations 1 and 2 (section 5.3.1) the output APLL is in Block A and is always locked to the DPLL as shown in [Figure 5. APLLCR3.APLLMUX=111](#). The Block B APLL is used for master clock multiplication.

In top-level configuration 3 (section 5.3.2) the output APLL is in Block B and is always locked to the jitter attenuator ([APLLCR3.APLLMUX=111](#)), which in turn is locked to the DPLL in Block A as shown in [Figure 6](#). The Block A APLL is used for master clock multiplication.

5.9.2 APLL Output Frequency

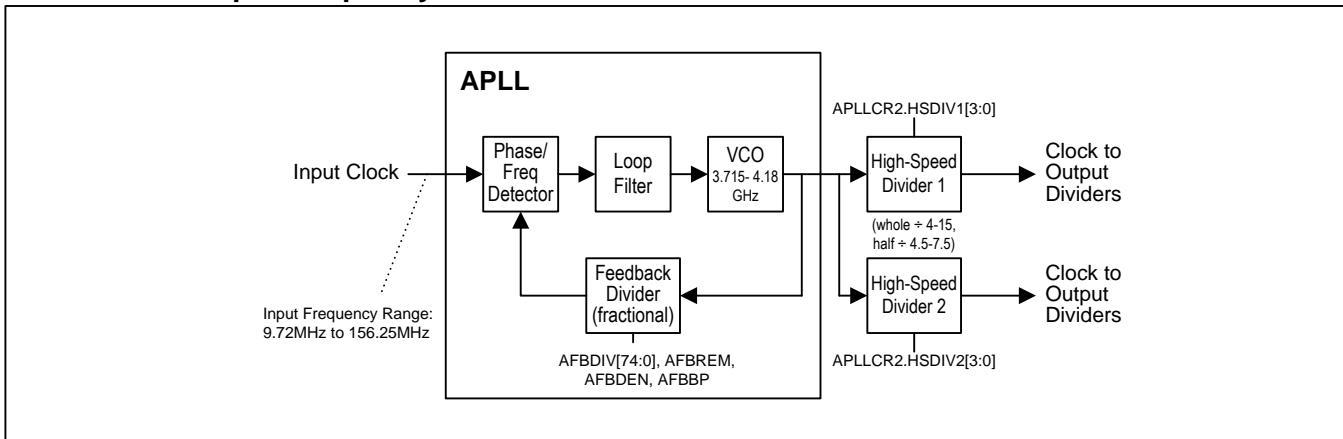


Figure 10 - APLL Block Diagram

The APLL is enabled when [PLLEN.APLLEN=1](#). The APLL has a fractional-N architecture and therefore can produce output frequencies that are either integer or non-integer multiples of the input clock frequency. [Figure 10](#) shows a block diagram of the APLL, which is built around an ultra-low-jitter multi-GHz VCO. Register fields [AFBDIV](#), [AFBREM](#), [AFBDEN](#) and [AFBBP](#) configure the frequency multiplication ratio of the APLL. The [APLLCR2.HSDIV1](#) and [HSDIV2](#) fields specify how the VCO frequency is divided down by the high-speed dividers. Dividing by six is the typical setting to produce 622.08MHz for SDH/SONET or 625MHz for Ethernet applications.

Internally, the exact APLL feedback divider value is expressed in the form [AFBDIV](#) + [AFBREM](#) / [AFBDEN](#) * $2^{(33-\text{AFBBP})}$. This feedback divider value must be chosen such that [APLL_input_frequency * feedback_divider_value](#) is in the operating range of the VCO (as specified in [Table 13](#)). The [AFBDIV](#) term is a fixed-point number with 7 integer bits and a configurable number of fractional bits (up to 33, as specified by [AFBBP](#)). Typically [AFBBP](#) is set to 9 to specify that [AFBDIV](#) has $33 - 9 = 24$ fractional bits. Using more than 24 fractional bits does not yield a detectable benefit. Using less than 12 fractional bits is not recommended.

The following equations show how to calculate the feedback divider values for the situation where the APLL should multiply the APLL input frequency by integer M and also fractionally scale by the ratio of integers N / D. In other words, $\text{VCO_frequency} = \text{input_frequency} * M * N / D$. An example of this is multiplying 77.76MHz from the DPLL by M=48 and scaling by N / D = 255 / 237 for forward error correction applications.

$$\text{AFBDIV} = \text{trunc}(M * N / D * 2^{24}) \quad (1)$$

$$\text{lsb_fraction} = M * N / D * 2^{24} - \text{AFBDIV} \quad (2)$$

$$\text{AFBDEN} = D \quad (3)$$

$$\text{AFBREM} = \text{round}(\text{lsb_fraction} * \text{AFBDEN}) \quad (4)$$

$$\text{AFBBP} = 33 - 24 = 9 \quad (5)$$

The [trunc\(\)](#) function returns only the integer portion of the number. The [round\(\)](#) function rounds the number to the nearest integer. In Equation (1), [AFBDIV](#) is set to the full-precision feedback divider value, $M * N / D$, truncated after the 24th fractional bit. In Equation (2) the temporary variable 'lsb_fraction' is the fraction that was truncated in Equation (1) and therefore is not represented in the [AFBDIV](#) value. In Equation (3), [AFBDEN](#) is set to the denominator of the original $M * N / D$ ratio. In Equation (4), [AFBREM](#) is calculated as the integer numerator of a fraction (with denominator [AFBDEN](#)) that equals the 'lsb_fraction' temporary variable. Finally, in Equation (5) [AFBBP](#) is set to $33 - 24 = 9$ to correspond with [AFBDIV](#) having 24 fractional bits.

When a fractional scaling scenario involves multiplying an integer M times multiple scaling ratios N_1 / D_1 through N_n / D_n , the equations above can still be used if the numerators are multiplied together to get $N = N_1 \times N_2 \times \dots \times N_n$ and the denominators are multiplied together to get $D = D_1 \times D_2 \times \dots \times D_n$.

The easiest way to calculate the exact values to write to the APLL registers is to use the evaluation software, available on the Microsemi website. This software can be used even when no evaluation board is attached to the computer.

Note: After the APLL's feedback divider settings are configured in register fields [AFBDIV](#), [AFBREM](#), [AFBDEN](#) and [AFBBP](#), the APLL enable bit [PLLLEN](#).APLLEN should be changed from 0 to 1 to cause the APLL to reacquire lock with the new settings. The real-time lock/unlock status of the APLL is indicated by [APLLSR](#).ALK and ALK2.

5.9.3 APLL Phase Adjustment

The phase of the APLL's output clock can be incremented or decremented by 1/8th of a VCO cycle. This phase step size is 30ps at maximum VCO frequency of 4180MHz and 33.7ps at minimum VCO frequency of 3715MHz. The [APLLCR4](#).PDSS field specifies the phase decrement control signal, which can be the [APLLCR4](#).DECPH bit or any of the four GPIOs. The [APLLCR4](#).PISS field specifies the phase increment control signal, which can be the [APLLCR4](#).INCIPH bit or any of the four GPIOs. Phase is adjusted on every rising edge and every falling edge of the control signal. This phase adjustment affects the output of both high-speed dividers.

5.10 Output Clock Configuration

The device has three output clock signal pairs that are synchronized to the DPLL. In configurations 1 and 2 (section 5.3.1) these are the Block A outputs. In configuration 3 (section 5.3.2) these are the Block B outputs.

Each output has individual divider, enable and signal format controls. In CMOS mode each signal pair can become two CMOS outputs, allowing the device to have up to six output clock signals. Also in CMOS mode, the OCxN pin can have an additional divider allowing the OCxN frequency to be an integer divisor of the OCxP frequency (example: OC3P 125MHz and OC3N 25MHz). The outputs can be aligned relative to each other and relative to an input signal, and the phases of output signals can be adjusted dynamically with high resolution and infinite range.

5.10.1 Output Enable, Signal Format, Voltage and Interfacing

To use an output, the output driver must be enabled by setting [OCxCR2](#).OCSF≠0, and the per-output dividers must be enabled by setting the appropriate bit in the [OCEN](#) register. The per-output dividers include the medium-speed divider, the low-speed divider and the associated phase adjustment/alignment circuitry and start/stop logic.

Using the [OCxCR2](#).OCSF register field, each output pair can be disabled or configured as a CML output, an HSTL output, or one or two CMOS outputs. When an output is disabled it is high impedance, and the output driver is in a low-power state. In CMOS mode, the OCxN pin can be disabled, in phase or inverted vs. the OCxP pin. In CML mode the normal 800mV V_{OD} differential voltage is available as well as a half-swing 400mV V_{OD} . All of these options are specified by [OCxCR2](#).OCSF. The clock to the output driver can be inverted by setting [OCxCR2](#).POL=1. The CMOS/HSTL output driver can be set to any of four drive strengths using [OCxCR2](#).DRIVE.

Each output has its own power supply pin to allow CMOS or HSTL signal swing from 1.5V to 3.3V for glueless interfacing to neighboring components. If OCSF is set to HSTL mode then a 1.5V power supply voltage should be used to get a standards-compliant HSTL output. Note that differential (CML) outputs must have a power supply of 3.3V.

The differential outputs can be easily interfaced to LVDS, LVPECL, CML, HCSL, HSTL and other differential inputs on neighboring ICs using a few external passive components. See [Figure 22](#) for examples.

5.10.2 Output Frequency Configuration

The frequency of each output is determined by the configuration of the APLL, the high-speed dividers and the per-output dividers. Each output can be connected to either high-speed divider 1 (HSDIV1) or 2 (HSDIV2) using the [OCxCR3](#).DIVSEL field.

Each output has two output dividers, a 7-bit medium-speed divider ([OCxCR1.MSDIV](#)) and a 25-bit low-speed output divider (LSDIV field in the [OCxDIV](#) registers). These dividers are in series, medium-speed divider first then output divider. These dividers produce signals with 50% duty cycle for all divider values including odd numbers. The low-speed divider can only be used if the medium-speed divider is used (i.e. [OCxCR1.MSDIV>0](#)). The maximum input frequency to the medium-speed divider is 850MHz. The maximum input frequency to the low-speed divider is 425MHz.

Since each output has its own independent dividers, the device can output a family of related frequencies that have an APLL HSDIV output frequency as a common multiple. For example, for Ethernet clocks, a 625MHz HSDIV output clock can be divided by four for one output to get 156.25MHz, divided by five for another output to get 125MHz, and divided by 25 for another output to get 25MHz. Similarly, for SDH/SONET clocks, a 622.08MHz HSDIV output clock can be divided by 4 to get 155.52MHz, by 8 to get 77.76MHz, by 16 to get 38.88MHz or by 32 to get 19.44MHz.

Two Different Frequencies in 2xCMOS Mode

When an output is in 2xCMOS mode it can be configured to have the frequency of the OCxN clock be an integer divisor of the frequency of the OCxP clock. Examples of where this can be useful:

- 125MHz on OCxP and 25MHz on OCxN for Ethernet applications
- 77.76MHz on OCxP and 19.44MHz on OCxN for SONET/SDH applications
- 25MHz on OCxP and 1Hz (i.e. 1PPS) on OCxN for telecom applications with Synchronous Ethernet and IEEE1588 timing

An output can be configured to operate like this by setting the LSDIV value in the [OCxDIV](#) registers to OCxP_freq / OCxN_freq - 1 and setting [OCxCR3.LSSEL=0](#) and [OCxCR3.NEGLSD=1](#). Here are some notes about this dual-frequency configuration option:

- In this mode only the medium speed divider is used to create the OCxP frequency. The low-speed divider is then used to divide the OCxP frequency down to the OCxN frequency. This means that the lowest OCxP frequency is the high-speed divider output frequency divided by 128.
- An additional constraint is that the medium-speed divider must be configured to divide by 6 or more (i.e. must have [OCxCR1.MSDIV≥5](#)).

5.10.3 Output Duty Cycle Adjustment

For output frequencies less than or equal to 141.666MHz, the duty cycle of the output clock can be modified using the [OCxDC](#).OCDC register field. This behavior is only available when [MSDIV>0](#) and [LSDIV > 1](#). When [OCDC = 0](#) the output clock is 50%. Otherwise the clock signal is a pulse with a width of OCDC number of MSDIV output clock periods. The range of OCDC can create pulse widths of 1 to 255 MSDIV output clock periods. When [OCxCR2.POL=0](#), the pulse is high and the signal is low the remainder of the cycle. When [POL=1](#), the pulse is low and the signal is high the remainder of the cycle.

Note that duty cycle adjustment is done in the low-speed divider. Therefore when [OCxCR3.LSSEL=0](#) the duty cycle of the output is not affected. Also, when a CMOS output is configured with [OCxCR3.LSSEL=0](#) and [OCxCR3.NEGLSD=1](#), the OCxN pin has duty cycle adjustment but the OCxP pin does not. This allows a higher-speed 50% duty cycle clock signal to be output on the OCxP pin and a lower-speed frame/phase/time pulse (e.g. 2kHz, 8kHz or 1PPS) to be output on the OCxN pin at the same time.

An output configured for CMOS or HSTL signal format should not be configured to have a duty cycle with high time shorter than 2ns or low time shorter than 2ns.

5.10.4 Output Phase Adjustment and Phase Alignment

The device has flexible, high-resolution tools for managing the phases of the output clocks relative to one another. The key register fields for this are found in the [PACR1](#) and [PACR2](#) global configuration registers and the per-output [OCxPH](#) register.

Phase alignment and phase adjustment are done in the medium-speed dividers. Resolution is 0.5 periods (also known as unit intervals or UI) of the high-speed divider (HSDIV) output clock. For example, for an HSDIV output frequency of 800MHz, resolution is 625ps.

5.10.4.1 Phase Adjustment

A phase adjustment is a phase change for an output relative to that output's most recent phase. To cause the device to perform phase adjustment of an output clock, set [PACR1.MODE](#)=1, set [OCxCR1.PHEN](#)=1 to enable the output for phase adjustment, and write the phase adjustment amount to the output's [OCxPH](#) register. Then an arm/trigger methodology is used to cause the phase adjustment to happen.

The arm step tells the device that it is enabled to perform the phase adjustment when it sees the trigger stimulus. The source of the arm signal is specified by [PACR2.ARMSRC](#). Options include the 0-to-1 transition of the [PACR1.ARM](#) bit, APLL transition from unlocked to locked, DPLL transition from unlocked to locked, or a transition on one of the GPIO pins.

The source of the trigger signal is specified by [PACR2.TRGSRC](#). Options include 0-to-1 transition of the [PACR1.TRIG](#) bit, APLL transition from unlocked to locked, DPLL transition from unlocked to locked, a rising edge of the DPLL input clock, or a transition on one of the GPIO pins. The trigger signal can be inverted by setting [PACR1.TINV](#). With [TINV](#)=1, the same GPIO signal can arm on one edge and trigger on the opposite edge.

Any combination of a macro-block's outputs can be phase adjusted by the same trigger, and each output can be adjusted by a different amount. Only outputs with [OCxCR1.PHEN](#)=1 and [OCxPH.PHADJ](#)≠0 have their phases adjusted.

There are a few constraints on the range of possible phase adjustments. These have to do with the output's medium-speed divider value.

- 1) Phase adjustment is not available unless [OCxCR1.MSDIV](#)>0.
- 2) The largest negative phase adjustment magnitude in HSDIV periods is:
 - If [OCxCR1.MSDIV](#) is odd: $(OCxCR1.MSDIV - 1) / 2$
 - If [OCxCR1.MSDIV](#) is even: $(OCxCR1.MSDIV - 2) / 2$
- 3) The largest positive phase adjustment in HSDIV periods is:
 - If [OCxCR1.MSDIV](#) is odd: $(127 - OCxCR1.MSDIV) / 2$
 - If [OCxCR1.MSDIV](#) is even: $(128 - OCxCR1.MSDIV) / 2$

The implications of constraints 2) and 3) are shown in this table:

OCxCR1.MSDIV	Largest Negative Phase Adjust, HSDIV periods	Largest Positive Phase Adjust, HSDIV periods	Notes
1 or 2	0	63	no negative adjustment
3 or 4	1	62	
5 or 6	2	61	
...	
123 or 124	61	2	
125 or 126	62	1	
127	63	0	no positive adjustment

During a phase adjustment the MSDIV output period is changed for one period. The MSDIV output signal during that period will have longer high time (unless inverted) during a positive phase adjustment and shorter high time

(unless inverted) during a negative phase adjustment. With negative phase adjustments care must be taken to not shorten the high time of the output clock signal to be too short for the components that receive the clock. There are several possible ways to avoid this issue including: (1) using small negative adjustments such as -0.5UI repeatedly instead of one larger negative adjustment, (2) using positive adjustments to “wrap around” to the desired negative adjustment, or (3) holding the components that receive the clock in reset during the phase adjustment.

An armed phase adjustment can be canceled before the trigger occurs by setting the **PACR1.RST** bit.

The **PASR** register has real-time status bits indicating whether a phase adjustment is armed and waiting for a trigger (ARMED bit) or in progress (BUSY bit). It also has a latched status bit (ADJL bit) to indicate the adjustment has completed.

Example: +1.0 HSDIV period phase adjustment for output OC1 using TRIG register bit:

```
OC1CR1.PHEN=1      (Enable phase adjust on OC1)
OC1PH.PHADJ=00000010 (Specify +1.0 HSDIV period phase adjustment)
PACR1.MODE=1        (Phase adjustment mode)
PACR2.ARMSRC=0001  (arm signal is PACR1.ARM bit)
PACR2.TRGSR=0000   (trigger signal is PACR1.TRIG bit)
PACR1.RST=1         (reset phase adjust/align state machine after changing ARMSRC)
PACR1.ARM=1         (arm for phase adjust)
PACR1.TRIG=1        (do the phase adjust: add +1.0 UI to output phase)
repeat the next two writes as needed:
PACR1.ARM=1 .TRIG=0 (arm again; clearing the TRIG bit is required when MSDIV period < master
                      clock period because TRIG is not self-clearing in this situation)
PACR1.TRIG=1        (add +1.0 UI to output phase again)
```

5.10.4.2 Phase Alignment, Output-to-Output

A phase alignment is a special case of phase adjustment where the MSDIV and LSDIV dividers for all participating outputs are reset just before the phase adjustment occurs. For output-to-output alignment the trigger can be the **PACR1.TRIG** bit or the APLL or DPLL lock signals.

To avoid glitches (i.e. “runt pulses”) on the output clock it is possible to manually stop the output(s), before triggering the phase alignment, and then restart the output(s) after the alignment (See section 5.10.5).

When aligning outputs, it is important to note that, by default, the phase of outputs configured as HSTL format or “two CMOS, OCxP inverted vs. OCxN” format is opposite that of CML outputs. For example, consider the case where OC1 is 100MHz CML format and OC2 is 100MHz HSTL format. When OC1 and OC2 are aligned then OC2N is high when OC1P is high. The polarity bit **OCxCR2.POL** can be used to change this as needed.

5.10.5. There are several rules when alignment is enabled for multiple outputs:

- All participating outputs must come from the same high-speed divider
- All outputs that use both medium-speed and low-speed divider must have the same MSDIV value, the same LSDIV value and PHADJ=0. Subsequent phase adjustment(s) can be used to move the output(s) to other phase(s).
- All outputs that only use medium-speed divider can have PHADJ values smaller than the period of the highest output frequency among them.
- When some outputs use only medium-speed divider and other outputs use both medium-speed and low-speed divider, all MSDIV values must be the same, and those output using low-speed divider must have PHADJ=0.

Contact Microsemi Timing Applications Support for help with alignment scenarios that don't meet the rules listed above.

Example: OC1-to-OC2 alignment (+3.5 HSDIV UI offset) after the APLL locks:

```

OC1CR1.PHEN=1      (Enable phase adjust on OC1)
OC2CR1.PHEN=1      (Enable phase adjust on OC2)
OC1PH.PHADJ=00000000 (0.0UI)
OC2PH.PHADJ=00000111 (+3.5UI)
PACR1.MODE=0        (Phase alignment mode)
PACR2.ARMSRC=0001  (arm signal is PACR1.ARM bit)
PACR2.TRGSRC=0001  (trigger signal is APLL transition from unlocked to locked)
PACR1.RST=1        (reset phase adjust/align state machine after changing ARMSRC, TRGSRC)
PACR1.ARM=1        (arm for phase alignment)
(Aligns/realigns outputs when the APLL locks or relocks)

```

5.10.4.3 Phase Alignment, Input-to-Output

The phase alignment tool described in section [5.10.4.2](#) can use a GPIO pin as the alignment trigger. However there is some uncertainty associated with sampling the GPIO signal. Therefore the phase alignment tool *by itself* is not sufficient to achieve input-to-output phase alignment.

5.10.4.3.1 Automatic with External Feedback

To align output signals to an input signal, the best approach is to use external feedback in which an output is externally connected to one of the DPLL's inputs. To enable external feedback, set [ICSCR1.FBSEL](#) to specify the ICx input to use for external feedback and disable DPLL hitless switching. In this configuration the DPLL, in a closed-loop manner, automatically phase-aligns its OCx outputs to the DPLL's selected reference. Any small error in this alignment due to wire delays can be compensated in the DPLL's [DPHOFF](#) registers. Also, phase adjustment as described in section [5.10.4.1](#) can be used to change the phases of output clocks vs. the input clock phase as needed.

5.10.4.3.2 Manual with Phase Alignment, Phase Measurement and Phase Adjustment

If for some reason external feedback cannot be used, open-loop input-to-output phase alignment can be accomplished under software control. The procedure is to first do a phase alignment as described in section [5.10.4.2](#) but with a GPIO input as the trigger. Then the phase measurement tool described in section [5.10.6](#) can be used to determine the phase difference between an output signal and the input signal. Then phase adjustment as described in section [5.10.4.1](#) can be used to change the phase of one or more output signals to align with input signal phase.

It is important to note that, by default, outputs that only use the medium-speed divider have their rising edge aligned with the rising edge of the trigger signal. Meanwhile, outputs that use both the medium-speed and low-speed dividers have their rising edge aligned with the falling edge of the trigger signal. Per-output polarity bits ([OCxCR2.POL](#)) can be used to invert the polarity of output signals as needed so that all are rising-edge aligned or falling-edge aligned or any combination as needed.

5.10.5 Output Clock Start and Stop

Output clocks can be stopped high or low. One use for this behavior is to ensure "glitchless" output clock operation while the output is reconfigured or phase aligned with some other signal.

Each output has an [OCxSTOP](#) register with fields to control this behavior. The [OCxSTOP.MODE](#) field specifies whether the output clock signal stops high, stops low, or does not stop. The [OCxSTOP.SRC](#) field specifies the source of the stop signal. Options include the [OCxSTOP.STOP](#) bit, assertion of one of the GPIO pins, and the arming of a phase adjustment (which is indicated by [PASR.ARMED](#)).

When the stop mode is Stop High ([OCxSTOP.MODE](#)=01) and the stop signal is asserted, the output clock is stopped after the next rising edge of the output clock. When the stop mode is Stop Low ([OCxSTOP.MODE](#)=10) and the stop signal is asserted, the output clock is stopped after the next falling edge of the output clock. Internally the clock signal continues to toggle while the output is stopped. When the stop signal is deasserted, the output

clock resumes on the opposite edge that it stopped on. Low-speed output clocks can take long intervals before being stopped after the stop signal goes active. For example, a 1 Hz output could take up to 1 second to stop.

[OCxCR1.MSDIV](#) must be > 0 for this function to operate since [MSDIV=0](#) bypasses the start-stop circuits. Note that when [OCxCR3.NEGLSD=1](#) the start-stop logic is bypassed for the OCxN pin, and OCxN may not start/stop without glitches.

When [OCxCR2.POL=1](#) the output stops on the opposite polarity that is specified by the [OCxSTOP.MODE](#) field.

When [OCxCR2.STOPDIS=1](#) the output driver is disabled (high impedance) while the output clock is stopped.

Each output has a status register ([OCxSR](#)) with several stop/start status bits. The [STOPD](#) bit is a real-time status bit indicating stopped or not stopped. The [STOPL](#) bit is a latched status bit that is set when the output clock has stopped. The [STARTL](#) bit is a latched status bit that is set when the output clock has started.

5.10.6 A-to-B Phase Offset Measurement

The phase or time offset between two of a macro-block's signals (signal A and signal B) can be measured in units of a timebase clock. This capability can be used for several purposes, including:

- Keeping output clocks and low-speed output phase/time signals—such as frame sync, multiframe sync, or 1 pulse per second (1PPS) signals—aligned with input phase/time signals. The A-to-B measurement circuitry can detect phase changes in the input signal. Then the DPLL's phase adjustment capability and/or the output phase adjustment circuitry described in section [5.10.4](#) can be used to move phase(s) of output(s) to follow the input phase change.
- Keeping output clock signals and/or low-speed output phase/time signals aligned with one another. The A-to-B measurement circuitry can detect relative phase changes, and the phase adjustment circuitry described in section [5.10.4](#) can be used to move phase(s) of output(s) as needed.

The A and B signals can be any of a macro-block's ICx inputs, OCx outputs, or GPIO pins, as specified by [MABCR2.ASRC](#) and [MABCR3.BSRC](#). The timebase signal can be the external oscillator signal (or the output of the crystal driver circuit, optionally doubled by the clock doubler) or the output clock of any of the three medium-speed dividers (MSDIV1, MSDIV2, MSDIV3). The timebase signal is specified by [MABCR1.TBSRC](#).

A Block A signal cannot be compared to a Block B signal and vice versa. This means, for example, that outputs cannot be compared to input when using device configuration 3 (section [5.3.2](#)) because in that configuration inputs are Block A signals and outputs are Block B signals.

A new measurement is started by writing [MABCR1.START=1](#). Any previously started measurement must be completed before a new measurement is started. If a measurement has not finished it can be aborted by writing [MABCR1.RST=1](#) before starting a new measurement. The measurement is complete when [MABSR1.RDYL](#) is set.

Example: consider an SDH/SONET application where OC1 is a 19.44MHz output clock and OC2 is an 8kHz frame sync signal. The goal is to measure the phase offset of OC1 vs. OC2. If they are found to have a phase offset then the phase adjustment circuitry in section [5.10.4](#) can be used to slowly change the phase of OC1 to match the phase of OC2.

MABCR1.TBSRC=001	(MSDIV1 output clock is 311.04MHz = 3.2 ns period)
MABCR2.ASRC=10001	(OC2 8kHz sync signal)
MABCR3.BSRC=10000	(OC1 19.44MHz clock)
MABCR1.START=1	(Start measurement)
Wait for MABSR1.RDYL=1	(Measurement ready)
Read MABSR1.OVFL	(to see if the measurement is valid)
MABSR1.RDYL=1, MABSR1.OVFL=1	(clear latched status bits)
Read MEAS bits from MABSR1 and MABSR2	

If, for example, MEAS = 111 1111 1001 (-8) then the rising edge of OC1 (the 'B' signal) precedes the rising edge of OC2 (the 'A' signal) by 8 MSDIV1 output clock periods (25.7ns).

An A-to-B measurement is performed by sampling the A and B signals with the selected timebase clock and detecting the rising or falling edges to measure. The number of timebase clocks between the A and B edges is counted. If the counter doesn't overflow then the phase difference is reported in the MEAS field in [MABSR1](#) and [MABSR2](#). If the counter does overflow then [MABSR1.OVFL](#) is set and the value of MEAS is invalid.

While the measurement is in progress the [MABSR1.BUSY](#) bit is set to 1. When the measurement is complete [MABSR1.BUSY](#) is set to 0 and [MABSR1.RDYL](#) is set to 1. Since the A and B signals are sampled by the timebase signal, this measurement tool is only useful when the timebase signal is much higher frequency than the A and B signals (at least 8-10x). Also, when possible, the timebase signal frequency should be less than or equal to 1000 times faster than the the frequencies of the A and B frequencies to avoid measurement counter overflow.

Constraints on A-to-B measurement:

- $f_B = f_A \times N$ where f_A is the frequency of signal A, f_B is the frequency of signal B and N is a positive integer

When measuring from an ICx input or a GPIO (signal A) to an ICx or a GPIO (signal B) and when measuring from an OCx output to an OCx output, the measured value is MEAS * timebase_period. This measurement has a variability of 0 to +1 timebase clock period.

When measuring from an ICx input or a GPIO (signal A) to an OCx output (signal B), the measurement in time units is MEAS * timebase_period + 6 * HSCLK_period, where HSCLK_period is the period of the output of the high-speed divider from which OCx signal is derived. This measurement has a variability of 0 to +1 timebase clock period plus 0 to +1 HSCLK periods.

When measuring from an OCx output (signal A) to an ICx input or a GPIO (signal B), the the measurement in time units is MEAS * timebase_period - 6 * HSCLK_period, where HSCLK_period is the period of the output of the high-speed divider from which OCx signal is derived. This measurement has a variability of 0 to +1 timebase clock period plus 0 to +1 HSCLK periods.

Guidance for Use

When the A and B signals are aligned to within one timebase clock cycle, the measurement hardware does not report 0. Instead it reports a measurement value that is equivalent to +1 cycle of signal B.

If the timebase clock is \leq 1023 times faster than signal B (so that the MEAS field cannot overflow, unless signal B is grossly too slow or not toggling at all) then system software should check the measured phase value. If the measured value is equal to the period of signal B then the A and B signals are aligned.

If the timebase clock is 1024 to 2047 times faster than signal B (and therefore the measurement counter can overflow) then the measurement hardware reports overflow when the A and B signals are aligned to within one timebase clock cycle. This report of overflow can be distinguished from other overflow cases by setting [MABCR3.BINV=1](#) and then remeasuring from signal A to the opposite edge of signal B. If the new measured value is equal to half the period of signal B then the A and B signals are aligned.

If the timebase clock is $>$ 2047 times faster than signal B then the measurement hardware reports overflow when the A and B signals are aligned to within one timebase clock cycle. This report of overflow is not distinguishable from other overflow cases. One way system software could work around this to determine that A and B are aligned is to use phase adjustment to move one of the signals by 2 or more timebase clocks then remeasure. If the new measured value matches the phase adjustment then the signals were aligned before the phase adjustment. Software can then adjust the phase of the signal back to its original position. Not all applications can tolerate such phase adjustments; for those applications it is recommended that the timebase clock be \leq 2047 times faster than signal B.

5.11 Microprocessor Interface

The device can communicate over a SPI interface or an I²C interface.

Section 5.4 describes reset pin settings required to configure the device for these interfaces.

5.11.1 SPI Slave

SPI is a widely used master/slave bus protocol that allows a master and one or more slaves to communicate over a serial bus. SPI masters are typically microprocessors, ASICs or FPGAs. Data transfers are always initiated by the master, which also generates the SCLK signal. A device receives serial data on the MOSI (Master Out Slave In) pin and transmits serial data on the MISO (Master In Slave Out) pin. MISO is high impedance except when the device is transmitting data to the bus master.

Each macro-block (A and B) can present a SPI slave port on the CSN, SCLK, MOSI, and MISO pins. Typically the two blocks' SPI ports are wired to the same SPI bus as follows:

- SCLK_A to SCLK_B to SPI master's SCLK pin
- MOSI_A to MOSI_B to SPI master's MOSI pin
- MISO_A to MISO_B to SPI master's MISO pin
- CSN_A to one CSN (or GPIO) output on SPI master
- CSN_B to one CSN (or GPIO) output on SPI master

Bit Order. The register address and all data bytes are transmitted most significant bit first on both MOSI and MISO.

Clock Polarity and Phase. The device latches data on MOSI on the rising edge of SCLK and updates data on MISO on the falling edge of SCLK. SCLK does not have to toggle between accesses, i.e., when CSN is high.

Device Selection. Each macro-block has its own chip-select line. To select the block, the bus master drives its CSN pin low.

Command and Address. After driving CSN low, the bus master transmits an 8-bit command followed by a 16-bit register address. The available commands are shown below.

Table 4 – SPI Commands

Command	Hex	Bit Order, Left to Right
Write Enable	0x06	0000 0110
Write	0x02	0000 0010
Read	0x03	0000 0011
Read Status	0x05	0000 0101

Read Transactions. The device registers are accessible when **EESEL**=0. The internal EEPROM memory is accessible when **EESEL**=1. After driving CSN low, the bus master transmits the read command followed by the 16-bit address. The device then responds with the requested data byte on MISO, increments its address counter, and prefetches the next data byte. If the bus master continues to demand data, the device continues to provide the data on MISO, increment its address counter, and prefetch the following byte. The read transaction is completed when the bus master drives CSN high. See [Figure 11](#).

Register Write Transactions. The device registers are accessible when **EESEL**=0. After driving CSN low, the bus master transmits the write command followed by the 16-bit register address followed by the first data byte to be written. The device receives the first data byte on MOSI, writes it to the specified register, increments its internal address register, and prepares to receive the next data byte. If the master continues to transmit, the device continues to write the data received and increment its address counter. The write transaction is completed when the bus master drives CSN high. See [Figure 13](#).

EEPROM Writes. The EEPROM memory is accessible when **EESEL**=1. After driving CSN low, the bus master transmits the write enable command and then drives CSN high to set the internal write enable latch. The bus master then drives CSN low again and transmits the write command followed by the 16-bit address followed by the first data byte to be written. The device first copies the page to be written from EEPROM to its page buffer. The device then receives the first data byte on MOSI, writes it to its page buffer, increments its internal address register, and prepares to receive the next data byte. If the master continues to transmit, the device continues to write the data received to its page buffer and continues to increment its address counter. The address counter rolls over at the 32-byte page boundary (i.e. when the five least-significant address bits are 11111). When the bus master drives CSN high, the device transfers the data in the page buffer to the appropriate page in the EEPROM memory. See [Figure 12](#) and [Figure 13](#).

EEPROM Read Status. After the bus master drives CSN high to end an EEPROM write command, the EEPROM memory is not accessible for up to 5ms while the data is transferred from the page buffer. To determine when this transfer is complete, the bus master can use the Read Status command. After driving CSN low, the bus master transmits the Read Status command. The device then responds with the status byte on MISO. In this byte, the least significant bit is set to 1 if the transfer is still in progress and 0 if the transfer has completed.

Early Termination of Bus Transactions. The bus master can terminate SPI bus transactions at any time by pulling CSN high. In response to early terminations, the device resets its SPI interface logic and waits for the start of the next transaction. If a register write transaction is terminated prior to the SCLK edge that latches the least significant bit of a data byte, the data byte is not written. If an EEPROM write transaction is terminated prior to the SCLK edge that latches the least significant bit of a data byte, none of the bytes in that write transaction are written.

Design Option: Wiring MOSI and MISO Together. Because communication between the bus master and the device is half-duplex, the MOSI and MISO pins can be wired together externally to reduce wire count. To support this option, the bus master must not drive the MOSI/MISO line when the device is transmitting.

AC Timing. See [Table 19](#) and [Figure 24](#) for AC timing specifications for the SPI interface.

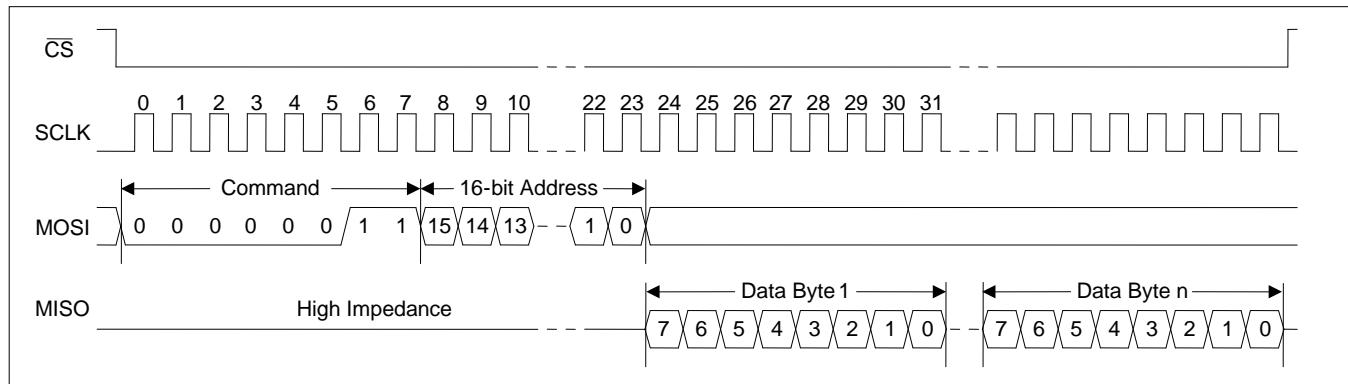


Figure 11 - SPI Read Transaction Functional Timing

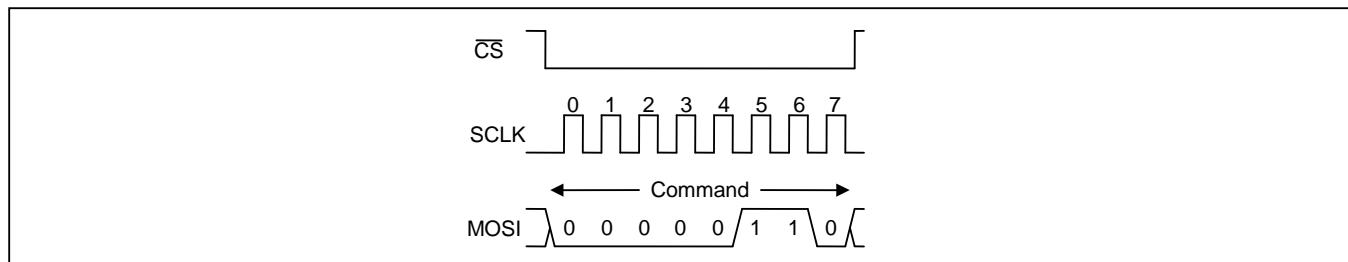


Figure 12 - SPI Write Enable Transaction Functional Timing

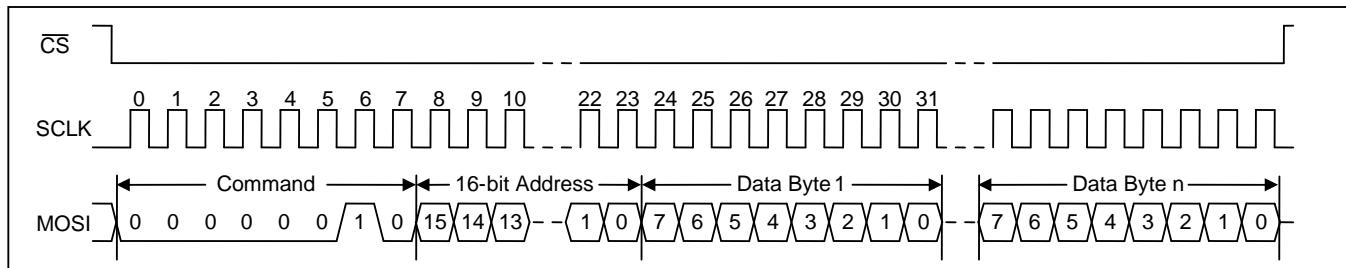


Figure 13 - SPI Write Transaction Functional Timing

5.11.2 I²C Slave

I²C is a widely used master/slave bus protocol that allows one or more masters and one or more slaves to communicate over a two-wire serial bus. I²C masters are typically microprocessors, ASICs or FPGAs. Data transfers are always initiated by the master, which also generates the SCL signal. The device is compliant with version 2.1 of the I²C specification.

The I²C interface on the device is a protocol translator from external I²C transactions to internal SPI transactions. This explains the slightly increased protocol complexity described in the paragraphs that follow.

Each macro-block (A and B) can present a fast-mode (400kbit/s) I²C slave port on the SCL and SDA pins. Typically the two blocks' I²C ports are wired to the same I²C bus as follows:

- SCL_A to SCL_B to I²C master's SCL pin
- SDA_A to SDA_B to I²C master's SDA pin
- IF0_A and IF0_B both wired or pulled to the same logic level
- IF1_A and IF1_B both wired or pulled to the same logic level

Read Transactions. The device registers are accessible when [EESEL=0](#). The internal EEPROM memory is accessible when [EESEL=1](#). The bus master first does an I²C write to the device. In this transaction three bytes are written: the SPI Read command (see [Table 4](#)), the upper byte of the register address, and the lower byte of the register address. The bus master then does an I²C read. During each acknowledge (A) bit the device fetches data from the read address and then increments the read address. The device then transmits the data to the bus master during the next 8 SCL cycles. The bus master terminates the read with a not-acknowledge (NA) followed by a STOP condition (P). See [Figure 14](#). Note: If the I²C write is separated from the I²C read by other I²C transactions then the device only outputs the data value from the first address and repeats that same data value after each acknowledge (A) generated by the bus master.

Register Write Transactions. The device registers are accessible when [EESEL=0](#). The bus master does an I²C write to the device. The first three bytes of this transaction are the SPI Write command (see [Table 4](#)), the upper byte of the register address, and the lower byte of the register address. Subsequent bytes are data bytes to be written. After each data byte is received, the device writes the byte to the write address and then increments the write address. The bus master terminates the write with a STOP condition (P). See [Figure 15](#).

EEPROM Writes. The EEPROM memory is accessible when [EESEL=1](#). The bus master first does an I²C write to transmit the SPI Write Enable command (see [Table 4](#)) to the device. The bus master then does an I²C write to transmit data to the device as described in the Register Write Transactions paragraph above. See [Figure 16](#).

EEPROM Read Status. The bus master first does an I²C write to transmit the SPI Read Status command (see [Table 4](#)) to the device. The bus master then does an I²C read to get the status byte. In this byte, the least significant bit is set to 1 if the transfer is still in progress and 0 if the transfer has completed. See [Figure 17](#).

I²C Features Not Supported by the Device. The I²C specification has several optional features that are not supported by the device. These are: 3.4Mbit/s high-speed mode (Hs-mode), 10-bit device addressing, general call address, software reset, and device ID. The device does not hold SCL low to force the master to wait.

I²C Slave Address. The two macro-blocks (A and B) have different I²C slave addresses. Three different combinations of slave addresses can be pin-configured as shown in the table in section 5.4.

Bit Order. The I²C specification requires device address, register address and all data bytes to be transmitted most significant bit first on the SDA signal.

Note: as required by the I²C specification, when power is removed from the device, the SDA and SCL pins are left floating so they don't obstruct the bus lines.

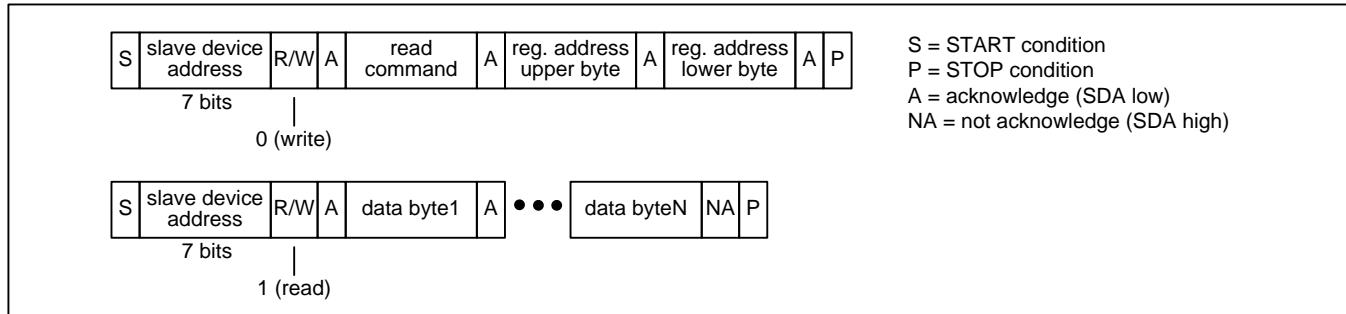


Figure 14 – I²C Read Transaction Functional Timing

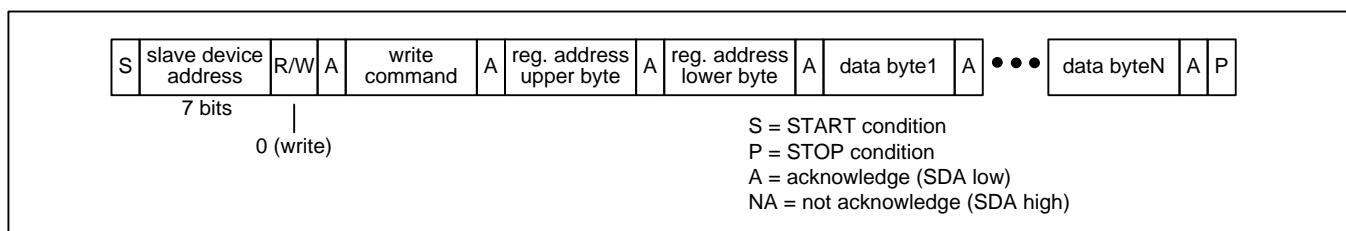


Figure 15 – I²C Register Write Transaction Functional Timing

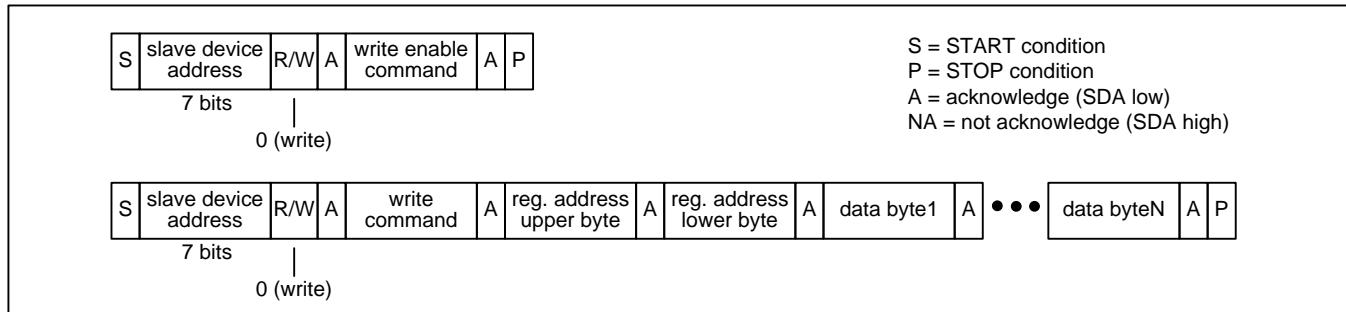


Figure 16 – I²C EEPROM Write Transaction Functional Timing

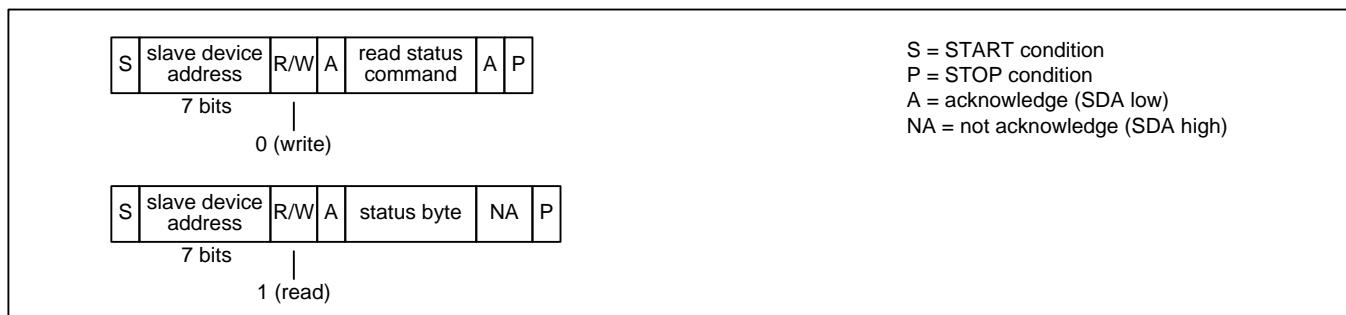


Figure 17 – I²C EEPROM Read Status Transaction Functional Timing

Note: In [Figure 14](#) through [Figure 17](#), a STOP condition (P) immediately followed by a START condition (S) can be replaced by a repeated START condition (Sr) as described in the I²C specification.

5.12 Interrupt Logic

Any of a macro-block's otherwise-unused GPIO pins can be configured as an interrupt-request output by setting the appropriate GPIOxC field in the [GPIOCR](#) registers to one of the status output options (01xx) and configuring the appropriate [GPIOxSS](#) register to follow the [INTSR.INT](#) bit. If system software is written to poll rather than receive interrupt requests, then software can read the [INTSR.INT](#) bit first to determine if any interrupt requests are active in the macro-block.

Many of the latched status bits in the device can be the source of an interrupt request if their corresponding interrupt enable bits are set. Each macro-block's interrupt logic is shown in [Figure 18](#). See the register map ([Table 5](#)) and the status register descriptions in section [6.3.2](#) for descriptions of the register bits shown in the figure.

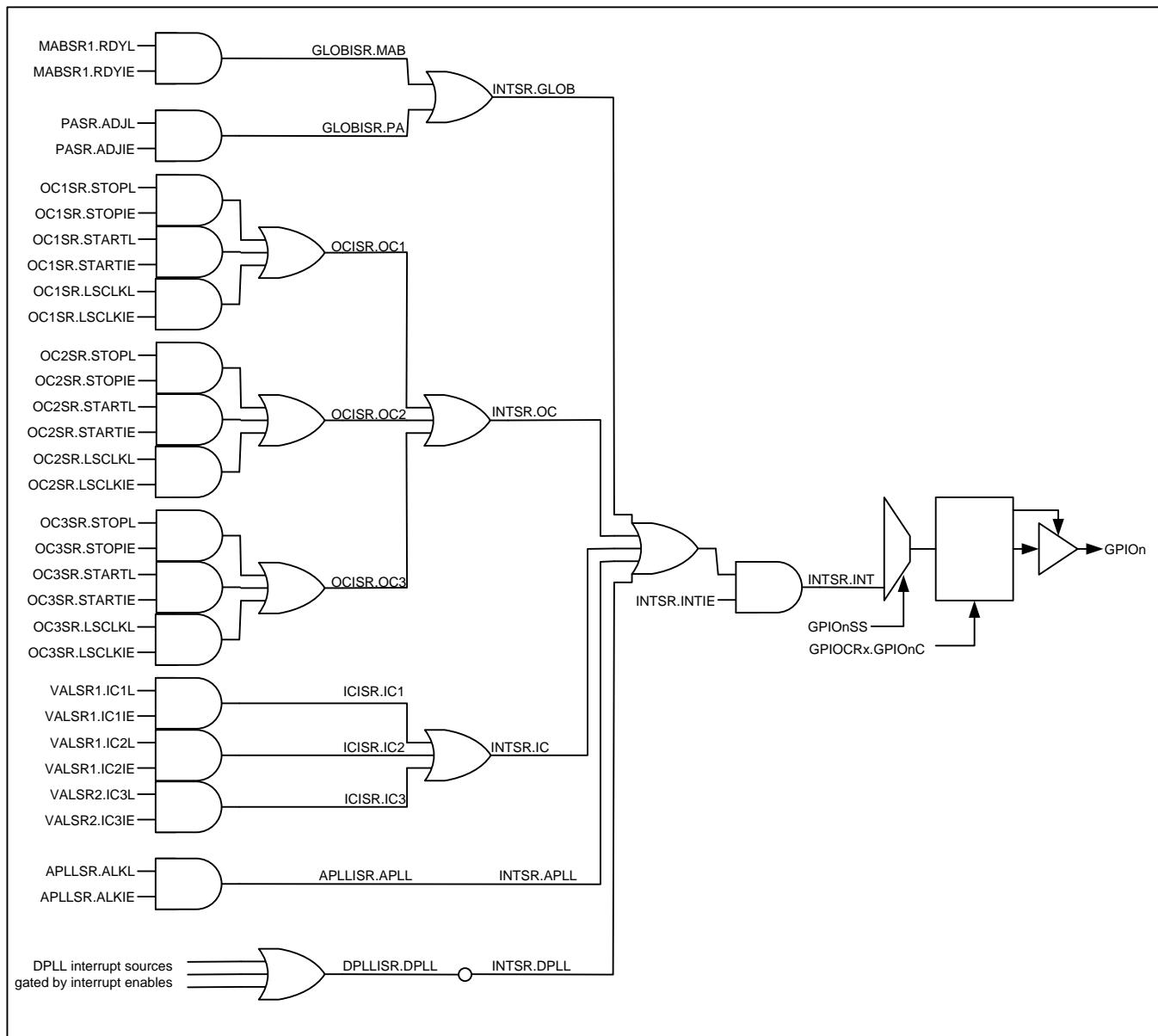


Figure 18 – Interrupt Structure

5.13 Reset Logic

The device has two reset controls: the RSTN pin and the RST bit in [MCR1](#). The RSTN pin asynchronously resets the entire device. When the RSTN pin is low all internal registers are reset to their default values. **The RSTN pin must have one rising edge after power-up.** At initial power-up reset should be asserted for at least 1 μ s. During operation, the RSTN assertion time can be as short as 1 μ s with one important exception:

Consider each of these four pins: AC0/GPIO0, AC1/GPIO1, TEST/GPIO2 and IF1/MISO. If (1) the pin could be an output driving high when RSTN is asserted, and (2) an external pulldown resistor is used to set the at-reset value of the pin, then RSTN should be asserted for 100 milliseconds.

The [MCR1.RST](#) bit resets the entire device (except for the microprocessor interface and the RST bit itself), but when the RST bit is active, the register fields with pin-programmed defaults do not latch their values from, or based on, the corresponding input pins. Instead these fields are reset to the default values that were latched when the RSTN pin was last active.

Microsemi recommends holding RSTN low while the internal ring oscillator starts up and stabilizes. An incorrect reset condition could result if RSTN is released before the oscillator has started up completely. After the external oscillator or internal crystal driver circuit has been enabled and stabilized, the master clock can be switched from the ring oscillator to the external oscillator using the [MCR1.MCSEL](#) bit.

Important: System software must wait at least 100 μ s after RSTN is deasserted and wait for [GLOBISR.BCDONE=1](#) before configuring the device.

5.14 Power-Supply Considerations

Due to the multi-power-supply nature of the device, some I/Os have parasitic diodes between a <3.3V supply and a 3.3V supply. When ramping power supplies up or down, care must be taken to avoid forward-biasing these diodes because it could cause latchup. Two methods are available to prevent this. The first method is to place a Schottky diode external to the device between the <3.3V supply and the 3.3V supply to force the 3.3V supply to be within one parasitic diode drop of the <3.3V supply. The second method is to ramp up the 3.3V supply first and then ramp up the <3.3V supply. In some applications VDDOx power supply pins can be at other voltages, such as 2.5V or 1.5V. In these applications the general solution is to ramp up the supplies in order from highest nominal to lowest nominal voltage.

5.15 Auto-Configuration from EEPROM

Each block's internal EEPROM memory can store up to four configurations, known as configurations 0, 1, 2 and 3. As described in section [5.4](#), the configuration to be used is specified by the values of the AC[1:0] pins at reset.

5.15.1 Generating Device Configurations

Device configurations must be generated using the evaluation software. This is true for auto-configurations stored in internal or external EEPROM and for configurations that are written to the device by a system processor. The reason for this requirement is that writes to undocumented registers must be done to tune analog circuitry for optimal performance. The writes to be done depend on integer vs. fractional multiplication, device mode and other factors. The registers involved control very low-level device parameters that are difficult to describe and difficult to understand how to use. Instead the evaluation software has all of the expert knowledge built-in to keep configuration easy for the user.

5.15.2 Direct EEPROM Write Mode

To simplify writing the internal EEPROM during manufacturing, the device has a test mode known as direct EEPROM write mode. The device enters this mode when TEST=1 and AC[1:0]=00 on the rising edge of RSTN. In this mode the EEPROM memory is mapped into the address map and can be written as needed to store configuration scripts in the device. Device registers are not accessible in this mode. The device exits this mode

when TEST=0 on the rising edge of RSTN. Note: the device drives the MISO pin continually during this mode. Therefore this mode cannot be used (a) when MOSI and MISO are tied together (as described in the *Design Option: Wiring MOSI and MISO Together* paragraph in section 5.11.1) and/or (b) when the MISO pins of the two blocks are tied together.

5.15.3 Holding Other Devices in Reset During Auto-Configuration

Using the appropriate `GPIOCR` and `GPIO0SS` registers, a GPIO pin can be configured to follow the `GLOBISR.BCDONE` status bit. This GPIO can then be used as a reset signal to hold other devices (that use clocks from this device) in reset while the device configures itself. For completeness one GPIO from each of the two blocks should be wire-ORed together with an external pullup resistor to make a signal that only goes high when both blocks are done with auto-configuration. To do this on GPIO0, for example, add the following writes at the beginning of each block's configuration script: write 0x1F to `GPIO0SS` and write 0x06 to `GPIOCR1`. In configuration files, block B's registers are address offset 0x4000.

5.16 Power Supply Decoupling and Layout Recommendations

Application Note ZLAN-524 describes recommended power supply decoupling and layout practices.

6. Register Descriptions

Each macro-block (A and B) has an overall address range from 000h to 6FFh. Table 5 shows the register map. In each register, bit 7 is the MSb and bit 0 is the LSb. Register addresses not listed and bits marked “—” are reserved and must be written with 0. Writing other values to these registers may put the in a factory test mode resulting in undefined operation. Bits labeled “0” or “1” must be written with that value for proper operation. Register fields with underlined names are read-only fields; writes to these fields have no effect. All other fields are read-write. Register fields are described in detail in the register descriptions that follow Table 5.

6.1 Register Types

6.1.1 Status Bits

The device has two types of status bits. Real-time status bits are read-only and indicate the state of a signal at the time it is read. Latched status bits are set when a signal changes state (low-to-high, high-to-low, or both, depending on the bit) and cleared when written with a logic 1 value. Writing a 0 has no effect. When set, some latched status bits can cause an interrupt request if enabled to do so by corresponding interrupt enable bits. Status bits marked “—” are reserved and must be ignored.

6.1.2 Configuration Fields

Configuration fields are read-write. During reset, each configuration field reverts to the default value shown in the register definition. Configuration register bits marked “—” are reserved and must be written with 0.

6.1.3 Multiregister Fields

Multiregister fields—such as `FREQZ[39:0]` in registers `DFREQZ1` through `DFREQZ5`—must be handled carefully to ensure that the bytes of the field remain consistent. A write access to a multiregister field is accomplished by writing all the registers of the field in order from smallest address to largest. Writes to registers other than the last register in the field (i.e. the register with the largest address) are stored in a transfer register. When the last register of the field is written, the entire multiregister field is updated simultaneously from the transfer register. If the last register of the field is not written, the field is not updated. Any reads from the multiregister field that occur during the middle of the multiregister write will read the existing value of the field not the new value in the transfer register.

A read access from a multiregister field is accomplished by reading the registers of the field in order from smallest address to largest. When the first register in the field (i.e. the register with the lowest address) is read, the entire multiregister field is copied to the transfer register. During subsequent reads from the other registers in the

multiregister field, the data comes from the transfer register. Any writes to the multiregister field that occur during the middle of the multiregister read will overwrite values in the transfer register.

The device has one write transfer register and one read transfer register that it reuses for all multiregister fields. For proper operation system software should be organized such that only one software process accesses the device's registers. If two or more processes are allowed to make uncoordinated accesses to the device's registers, their accesses to multiregister fields could interrupt one another leading to incorrect writes and reads of the multiregister fields.

The multiregister fields are:

Field	Registers	Type
FREQZ[39:0]	DFREQZ1 to DFREQZ5	Read/Write
PHOFF[31:0]	DPHOFF1 to DPHOFF4	Read/Write
FREQ[31:0]	DFREQ1 to DFREQ4	Read-Only
PHASE[31:0]	DPHASE1 to DPHASE4	Read-Only

6.1.4 Bank-Switched Registers

The **EESEL** register is a bank-select control field that maps the macro-block's registers into the memory map at address 0x1 and above when EESEL=0 and maps the EEPROM memory into the memory map at address 0x1 and above when EESEL=1. The EESEL register itself is always in the memory map at address 0x0 for both EESEL=0 and EESEL=1.

6.1.5 DPLL Registers

See section [5.8.1](#) for important discussion about DPLL registers.

6.2 Register Map

Each macro-block has the following register map. Each block has its own SPI/I²C interface through which its registers are accessed (see section [5.11](#)).

Table 5 - Register Map

ADDR	REGISTER	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
Global Configuration Registers									
00h	EESEL	—	—	—	—	—	—	—	EESEL
09	MCR1	RST	—	MCSEL1	MCSEL	ROSCD	DBL	XAB[1:0]	
0A	MCR2	—	EXTSS[1:0]		AINCDIS	—	MCLK[2:0]		
0B	PLLEN	—	—	—	DPLLEN	—	—	—	APLLEN
0C	ICEN	—	—	—	—	IC3EN	IC2EN	IC1EN	
0D	OCEN	—	—	—	—	OC3EN	OC2EN	OC1EN	
0E	GPIOCR1	GPIO1C[3:0]				GPIO0C[3:0]			
0F	GPIOCR2	GPIO3C[3:0]				GPIO2C[3:0]			
12	GPIO0SS	REG[4:0]					BIT[2:0]		
13	GPIO1SS	REG[4:0]					BIT[2:0]		
14	GPIO2SS	REG[4:0]					BIT[2:0]		
15	GPIO3SS	REG[4:0]					BIT[2:0]		
1B	PACR1	RST	TRIG	ARM	—	—	—	TINV	MODE
1C	PACR2	ARMSRC[3:0]				TRGSRC[3:0]			
1D	MABCR1	RST	START	—	—	—	TBSRC[2:0]		

ADDR	REGISTER	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0					
1E	MABCR2	AINV	—	—	ASRC[4:0]									
1F	MABCR3	BINV	—	—	BSRC[4:0]									
Status Registers														
30	ID1	<u>IDU[7:0]</u>												
31	ID2	<u>IDL[3:0]</u>				<u>REV[3:0]</u>								
40	CFGSR	<u>TEST</u>	<u>XOFAIL</u>	—	—	<u>IF[1:0]</u>	<u>AC[1:0]</u>							
41	GPIOSR	—	—	—	—	<u>GPIO3</u>	<u>GPIO2</u>	<u>GPIO1</u>	<u>GPIO0</u>					
42	INTSR	—	<u>GLOB</u>	<u>OC</u>	<u>IC</u>	<u>DPLL</u>	<u>APLL</u>	<u>INTIE</u>	<u>INT</u>					
43	GLOBISR	<u>BCDONE</u>	—	—	—	—	—	<u>PA</u>	<u>MAB</u>					
44	ICISR	—	—	—	—	—	<u>IC3</u>	<u>IC2</u>	<u>IC1</u>					
45	OCISR	—	—	—	—	—	<u>OC3</u>	<u>OC2</u>	<u>OC1</u>					
46	APLLISR	—	—	—	—	—	—	—	<u>APLL</u>					
47	DPLLISR	—	—	—	—	—	—	—	<u>DPLL</u>					
48	APLLSR	—	<u>ALK2IE</u>	<u>ALK2L</u>	<u>ALK2</u>	<u>ADLK</u>	<u>ALKIE</u>	<u>ALKL</u>	<u>ALK</u>					
49	PTAB1	—	—	<u>REF1[1:0]</u>		—	—	<u>SELREF[1:0]</u>						
4A	PTAB2	—	—	<u>REF3[1:0]</u>		—	—	<u>REF2[1:0]</u>						
4B	MABSR1	—	<u>OVFL</u>	<u>RDYIE</u>	<u>RDYL</u>	<u>BUSY</u>	<u>MEAS[10:8]</u>							
4C	MABSR2	<u>MEAS[7:0]</u>												
4D	PASR	—	—	—	—	<u>ADJIE</u>	<u>ADJL</u>	<u>BUSY</u>	<u>ARMED</u>					
4E	VALSR1	—	<u>IC2IE</u>	<u>IC2L</u>	<u>IC2</u>	—	<u>IC1IE</u>	<u>IC1L</u>	<u>IC1</u>					
4F	VALSR2	—	—	—	—	—	<u>IC3IE</u>	<u>IC3L</u>	<u>IC3</u>					
50	IC1SR	<u>LKCLR</u>	<u>LKTO</u>	<u>LOS</u>	<u>ACVAL</u>	<u>PCVAL</u>	<u>PPVAL</u>	<u>VAL</u>	<u>VALT</u>					
51	IC2SR	<u>LKCLR</u>	<u>LKTO</u>	<u>LOS</u>	<u>ACVAL</u>	<u>PCVAL</u>	<u>PPVAL</u>	<u>VAL</u>	<u>VALT</u>					
52	IC3SR	<u>LKCLR</u>	<u>LKTO</u>	<u>LOS</u>	<u>ACVAL</u>	<u>PCVAL</u>	<u>PPVAL</u>	<u>VAL</u>	<u>VALT</u>					
53	OC1SR	<u>LSCLKIE</u>	<u>LSCLKL</u>	<u>LSCLK</u>	<u>STARTIE</u>	<u>STARTL</u>	<u>STOPIE</u>	<u>STOPL</u>	<u>STOPD</u>					
54	OC2SR	<u>LSCLKIE</u>	<u>LSCLKL</u>	<u>LSCLK</u>	<u>STARTIE</u>	<u>STARTL</u>	<u>STOPIE</u>	<u>STOPL</u>	<u>STOPD</u>					
55	OC3SR	<u>LSCLKIE</u>	<u>LSCLKL</u>	<u>LSCLK</u>	<u>STARTIE</u>	<u>STARTL</u>	<u>STOPIE</u>	<u>STOPL</u>	<u>STOPD</u>					
56	DSRR1	—	—	<u>LOL</u>	<u>TRK</u>	<u>HO</u>	<u>FLOCK</u>	—	—					
57	DSRR2	—	—	—	<u>NORUN</u>	—	<u>FLIM</u>	—	<u>PSLIM</u>					
59	DSRL1	—	—	<u>LOLL</u>	<u>TRKL</u>	<u>HOL</u>	<u>FLOCKL</u>	—	—					
5A	DSRL2	—	—	—	—	—	<u>FLIML</u>	—	<u>PSLIML</u>					
5B	DSRIE1	—	—	<u>LOLIE</u>	<u>TRKIE</u>	<u>HOIE</u>	<u>FLOCKIE</u>	—	—					
5C	DSRIE2	—	—	—	—	—	<u>FLIMIE</u>	—	<u>PSLIMIE</u>					
65	DFREQ1	<u>FREQ[7:0]</u>												
66	DFREQ2	<u>FREQ[17:8]</u>												
67	DFREQ3	<u>FREQ[23:16]</u>												
68	DFREQ4	<u>FREQ[31:24]</u>												
7D	DPHASE1	<u>PHASE[7:0]</u>												
7E	DPHASE2	<u>PHASE[15:8]</u>												
7F	DPHASE3	<u>PHASE[23:16]</u>												
80	DPHASE4	<u>PHASE[31:24]</u>												
APLL Configuration Registers														
100	APLLCR1	—	—	—	—	—	<u>ENHS2</u>	<u>BYPHS2</u>	—					
101	APLLCR2	<u>HSDIV2[3:0]</u>				<u>HSDIV1[3:0]</u>								
102	APLLCR3	—	—	—	—	—	<u>APLLMUX[2:0]</u>							
103	APLLCR4	<u>DECPh</u>	<u>PDSS[2:0]</u>			<u>INCPH</u>	<u>PISS[2:0]</u>							
106	AFBDIV1	<u>AFBDIV[7:0]</u>												

ADDR	REGISTER	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
107	AFBDIV2								AFBDIV[15:8]
108	AFBDIV3								AFBDIV[23:16]
109	AFBDIV4								AFBDIV[31:24]
10A	AFBDIV5								AFBDIV[39:32]
10B	AFBDIV6	—	—	—	—	—	—	—	AFBDIV[41:40]
10C	AFBDEN1								AFBDEN[7:0]
10D	AFBDEN2								AFBDEN[15:8]
10E	AFBDEN3								AFBDEN[23:16]
10F	AFBDEN4								AFBDEN[31:24]
110	AFBREM1								AFBREM[7:0]
111	AFBREM2								AFBREM[15:8]
112	AFBREM3								AFBREM[23:16]
113	AFBREM4								AFBREM[31:24]
114	AFBBP								AFBBP[7:0]

Output Clock Configuration Registers

OC1 Registers															
200	OC1CR1	PHEN	MSDIV[6:0]												
201	OC1CR2	ASQUEL	POL	DRIVE[1:0]		STOPDIS	OCSF[2:0]								
202	OC1CR3	SRLSEN	DIVSEL	NEGLSD	LSSEL	—	—	—	LSDIV[24]						
203	OC1DIV1	LSDIV[7:0]													
204	OC1DIV2	LSDIV[15:8]													
205	OC1DIV3	LSDIV[23:16]													
206	OC1DC	OCDC[7:0]													
207	OC1PH	PHADJ[7:0]													
208	OC1STOP	STOP	—	SRC[3:0]			MODE[1:0]								
OC2 Registers															
210	OC2CR1	same as OC1 registers													
...	...														
218	OC2STOP														
OC3 Registers															
220	OC3CR1	same as OC1 registers													
...	...														
228	OC3STOP														

Input Clock Configuration

IC1 Registers												
300	IC1CR1	—	POL	—	—	—	—	HSDIV[1:0]				
311	MON1CR2	—	LOSSS[2:0]			—	ACEN	PCEN	PPEN			
IC2 Registers												
320	IC2CR1	same as IC1 registers										
331	MON2CR2											
IC3 Registers												
340	IC3CR1	same as IC1 registers										
351	MON3CR2											

DPLL Configuration Registers

400	ICSCR1	EXTSW	REVERT	—	FBSEL[1:0]		—	—	—
401	VALCR1	—	—	—	—	—	IC3	IC2	IC1
402	IPR1	—	—	PRI2[1:0]	—	—	—	PRI1[1:0]	

ADDR	REGISTER	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
403	IPR2	—	—	—	—	—	—	PRI3[1:0]	
404	PHLKTO	PHLKTO[1:0]				PHLKTO[5:0]			
405	LKATO	LKATOM[1:0]				LKATO[5:0]			
40B	DPLLCR1	—	—	HOMODE[1:0]			MODE[3:0]		
40C	DPLLCR2	—	—	—	—	—	—	HSEN[1:0]	
420	DFREQZ1				FREQZ[7:0]				
421	DFREQZ2				FREQZ[15:8]				
422	DFREQZ3				FREQZ[23:16]				
423	DFREQZ4				FREQZ[31:24]				
424	DFREQZ5				FREQZ[39:32]				
428	DPOFF1				PHOFF[7:0]				
429	DPOFF2				PHOFF[15:8]				
42A	DPOFF3				PHOFF[23:16]				
42B	DPOFF4				PHOFF[31:24]				

6.3 Register Definitions

6.3.1 Global Configuration Registers

Register Name: EESEL
Register Description: EEPROM Memory Selection Register
Register Address: 00h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	—	—	EESEL
Default	0	0	0	0	0	0	0	0

Bit 0: EEPROM Memory Select (EESEL). This bit is a bank-select that specifies whether register space or EEPROM memory is mapped into addresses 0x1 and above. See sections 5.11 and 6.1.4. Note that this bit is write-only; the value read is not reliable.

0 = Device registers

1 = EEPROM memory

Register Name: MCR1
Register Description: Master Configuration Register 1
Register Address: 09h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RST	—	MCSEL1	MCSEL	ROSCD	DBL	XAB[1:0]	
Default	0	0	0	0	0	0	0	0

Bit 7: Reset (RST). When this bit is high the entire device is held in reset, and all register fields, except the RST bit itself, are reset to their default states. When RST is high, the register fields with pin-programmed defaults do not latch their values from the corresponding input pins. Instead these fields are reset to the default values that were latched from the pins when the RSTN pin was last active. See section 5.13.

0 = Normal operation

1 = Reset

Note: For proper sequencing of internal logic, write MCR1 to clear the MCSEL1, MCSEL and ROSCD bits first (without changing the value of the RST bit) then perform a second write to set the RST bit.

Bit 5: DPLL Master Clock Select IC1 (MCSEL1). This bit overrides the MCSEL bit to specify IC1 as the source of the DPLL master clock.

0 = DPLL master clock selected by MCSEL bit

1 = DPLL master clock sourced from IC1, which has a divider and a polarity control bit

Bit 4: DPLL Master Clock Select (MCSEL). This bit selects the source of the DPLL master clock. At reset the internal ring oscillator is enabled and selected. For Block B when configured for jitter attenuation and always for Block A, this bit must be set to 1 after the external oscillator connected to the XA pin has stabilized and is ready to use. See section [5.5.3](#).

0 = DPLL master clock sourced from internal ring oscillator

1 = DPLL master clock sourced from the XA pin (optionally through the clock doubler)

Bit 3: Ring Oscillator Disable (ROSCD). This bit disables the ring oscillator. It can be set to 1 when either MCSEL or MCSEL1 is set 1 so that the ring oscillator does not cause unwanted phase noise spurs in output clock signals. See section [5.5.3](#).

0 = Enable

1 = Disable (power-down)

Bit 2: Clock Doubler Enable (DBL). (Block B only) This bit enables the clock doubler for the output of the crystal driver circuitry. During power-up, system software must wait at least 5ms for the crystal driver circuit to stabilize before enabling the clock doubler. The doubler can only be used with a crystal. It is not compatible with a clock signal from an XO or other source.

0 = Disable (power down)

1 = Enable

Bits 1 to 0: XA/XB Pin Mode (XAB[1:0]). This field specifies the behavior of the XA and XB pins. See section [5.5](#).

00 = Crystal driver and input disabled / powered down

01 = Crystal driver and input enabled on XA/XB

10 = XA enabled as single-ended input for external oscillator signal; XB must be left floating

11 = {unused value}

Register Name:

MCR2

Register Description:

Master Configuration Register 2

Register Address:

0Ah

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	EXTSS[1:0]		AINCDIS	—	MCLK[2:0]		
Default	0	0	0	0	0	0	0	0

Bits 6 to 5: External Switch Source Select (EXTSS[1:0]). (Block A only) This field selects the GPIO source for the DPLL external switch control signal. It is only valid when [ICSCR1.EXTSW=1](#). See sections [5.7.3.4](#).

00 = GPIO0

01 = GPIO1

10 = GPIO2

11 = GPIO3

Bit 4: Automatic Increment Disable (AINCDIS). This bit disables the automatic register address increment during SPI or I2C reads and writes. Disabling automatic increment significantly increases the efficiency of writing and reading DSP code, which requires repeated writes to address 0x040E or repeated reads from address 0x040F.

0 = Enable automatic address increment

1 = Disable automatic address increment

Bits 2 to 0: Master Clock Frequency Select (MCLK[2:0]). These bits specify the frequency range of the DPLL master clock. When the internal clock doubler is enabled: (a) this MCLK field must specify 2x the crystal frequency, and (b) the crystal must be in the 46.5MHz to 60MHz range.

00x = 93MHz to 104MHz

01x = 104MHz to 115MHz

10x = 115MHz to 130MHz

11x = {unused values}

Note that the master clock frequency range widens to 80MHz to 130MHz (with no gaps in the range) for applications where the DPLL is not expected to lock to input clock signals but rather is only operated in NCO (numerically controlled oscillator) mode. In these applications this MCLK field is ignored.

Register Name: PLLEN

Register Description: APLL and DPLL Enable Register

Register Address: 0Bh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	DPLLEN	—	—	—	APLLEN
Default	0	0	0	0	0	0	0	0

Bit 4: DPLL Enable (DPLLEN). This field enables or disables the DPLL and input clock monitors. The XA clock source must be properly configured and selected to operate the DPLL and input clock monitors.

0 = Disable (powered down)

1 = Enable

Bit 0: APLL Enable (APLLEN). This bit enables or disables the APLL. For normal operation the APLL must be enabled. See section [5.9.2](#).

0 = Disabled

1 = Enabled

Register Name: ICEN

Register Description: Input Clock Enable Register

Register Address: 0Ch

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	IC3EN	IC2EN	IC1EN
Default	0	0	0	0	0	0	0	0

Bit 2: Input Clock 3 Enable (IC3EN). This bit enables and disables the input clock 3 differential receiver and input divider. See section [5.6](#).

0 = Disabled

1 = Enabled

Bit 1: Input Clock 2 Enable (IC2EN). This bit enables and disables the input clock 2 differential receiver and input divider. See section [5.6](#).

0 = Disabled

1 = Enabled

Bit 0: Input Clock 1 Enable (IC1EN). This bit enables and disables the input clock 1 differential receiver and input divider. See section [5.6](#).

0 = Disabled

1 = Enabled

Register Name: OCEN
Register Description: Output Clock Enable Register
Register Address: 0Dh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	OC3EN	OC2EN	OC1EN
Default	0	0	0	0	0	0	0	0

Bit 2: Output Clock 3 Enable (OC3EN). This bit enables and disables the output clock 3 drivers, output dividers, phase adjustment/alignment circuitry and start/stop circuitry. See section [5.10.1](#).

0 = Disabled
1 = Enabled

Bit 1: Output Clock 2 Enable (OC2EN). This bit enables and disables the output clock 2 drivers, output dividers, phase adjustment/alignment circuitry and start/stop circuitry. See section [5.10.1](#).

0 = Disabled
1 = Enabled

Bit 0: Output Clock 1 Enable (OC1EN). This bit enables and disables the output clock 1 drivers, output dividers, phase adjustment/alignment circuitry and start/stop circuitry. See section [5.10.1](#).

0 = Disabled
1 = Enabled

Register Name: GPIOCR1
Register Description: GPIO Configuration Register 1
Register Address: 0Eh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	GPIO1C[3:0]					GPIO0C[3:0]		
Default	0	0	0	0	0	0	0	0

Bits 7 to 4: GPIO1 Configuration (GPIO1C[3:0]). This field configures the GPIO1 pin as a general-purpose input, a general-purpose output driving low or high, or a status output. The current state of the pin can be read from [GPIOSR.GPIO1](#). When GPIO1 is a status output, the [GPIO1SS](#) register specifies which status bit is output.

- 0000 = General-purpose input
- 0001 = General-purpose input - inverted polarity
- 0010 = General-purpose output driving low
- 0011 = General-purpose output driving high
- 0100 = Status output – non-inverted polarity
- 0101 = Status output – inverted polarity of the status bit it follows
- 0110 = Status output – 0 drives low, 1 high impedance
- 0111 = Status output – 0 high impedance, 1 drives low
- 1000 to 1111 = {unused values}

Bits 3 to 0: GPIO0 Configuration (GPIO0C[3:0]). This field configures the GPIO0 pin as a general-purpose input, a general-purpose output driving low or high, or a status output. The current state of the pin can be read from [GPIOSR.GPIO0](#). When GPIO0 is a status output, the [GPIO0SS](#) register specifies which status bit is output.

- 0000 = General-purpose input
- 0001 = General-purpose input - inverted polarity
- 0010 = General-purpose output driving low
- 0011 = General-purpose output driving high
- 0100 = Status output – non-inverted polarity
- 0101 = Status output – inverted polarity of the status bit it follows
- 0110 = Status output – 0 drives low, 1 high impedance
- 0111 = Status output – 0 high impedance, 1 drives low
- 1000 to 1111 = {unused values}

Note: in top-level configuration 3 (section [5.3.2](#)) Block A's GPIO0 has a specific required function, is configured by the evaluation software, and is not available as a GPIO.

Register Name: GPIOCR2
Register Description: GPIO Configuration Register 2
Register Address: 0Fh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	GPIO3C[3:0]					GPIO2C[3:0]		
Default	0	0	0	0	0	0	0	0

These fields are identical to those in [GPIOCR1](#) except they control GPIO2 and GPIO3.

Note: in top-level configuration 3 (section [5.3.2](#)) Block A's GPIO3 and Block B's GPIO3 have specific required functions, are configured by the evaluation software, and are not available as GPIOs.

Register Name: GPIO0SS
Register Description: GPIO0 Status Select Register
Register Address: 12h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name			REG[4:0]					
Default	0	0	0	0	0	0	0	0

Bits 7 to 3: Status Register (REG[4:0]). When [GPIOCR1.GPIO0C=01xx](#), this field specifies the register of the status bit that GPIO0 will follow while the BIT field below specifies the status bit within the register. Setting the combination of this field and the BIT field below to follow a bit that isn't implemented as a real-time or latched status register bit results in GPIO0 being driven low. The address of the status bit that GPIO0 follows is 0x40 + REG[4:0]

Bits 2 to 0: Status Bit (BIT[2:0]). When [GPIOCR1.GPIO0C=01xx](#), the REG field above specifies the register of the status bit that GPIO0 will follow while this field specifies the status bit within the register. Setting the combination of the REG field and this field to follow a bit that isn't implemented as a real-time or latched status register bit results in GPIO1 being driven low. 000=bit 0 of the register. 111=bit 7 of the register.

Note: The device does not allow status register bits [ICxSR.LOS](#) to be followed by a GPIO because the source of these bits could be the same GPIO. Also, the device does not allow the GPIO status register bits in [GPIOSR](#) to be followed by a GPIO.

Register Name: GPIO1SS
Register Description: GPIO1 Status Select Register
Register Address: 13h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name			REG[4:0]					
Default	0	0	0	0	0	0	0	0

These fields are identical to those in [GPIO0SS](#) except they control GPIO1.

Register Name: GPIO2SS
Register Description: GPIO2 Status Select Register
Register Address: 14h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name			REG[4:0]					
Default	0	0	0	0	0	0	0	0

These fields are identical to those in [GPIO0SS](#) except they control GPIO2.

Register Name: GPIO3SS
Register Description: GPIO3 Status Select Register
Register Address: 15h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name			REG[4:0]					
Default	0	0	0	0	0	0	0	0

These fields are identical to those in [GPIO0SS](#) except they control GPIO3.

Register Name: PACR1
Register Description: Phase Adjust Configuration Register 1
Register Address: 1Bh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RST	TRIG	ARM	—	—	—	TINV	MODE
Default	0	0	0	0	0	0	0	0

Block A for top-level configurations 1 and 2. Block B for top-level configuration 3.

Bit 7: Phase Adjustment Reset Bit (RST). This bit is used to reset the phase adjustment state machine. This is used to abort a phase adjustment after arming but before the trigger occurs. Resetting puts the state machine back to waiting for an arm signal. This bit is self-clearing. See section [5.10.4](#).

1 = Reset a phase adjustment event in progress, self clearing

Bit 6: Phase Adjustment Trigger Bit (TRIG). This bit is used to trigger the phase adjustment event when [PACR2.TRGSRC=0000](#) and the phase adjustment has been armed. This bit is self-clearing and must be written again to cause another trigger. When the ARM bit and TRIG bit are selected as the sources for arming and triggering, respectively, the ARM bit must be set first then the TRIG bit can be set in a subsequent register write to initiate a trigger event. See section [5.10.4](#).

1 = Trigger a phase adjustment, self clearing

Note: For phase adjustment when any OCx output's MSDIV period is less than master clock period, this bit may or may not self-clear depending on exact device configuration and therefore must be cleared by system software for proper operation.

Bit 5: Phase Adjustment Arm Bit (ARM). When [PACR2.ARMSRC=0001](#), setting this bit to 1 while [PASR.ARMED=0](#) arms the phase adjustment. Writing a 0 to this bit has no effect. Changing the value of this bit from 0 to 1 while [PASR.ARMED=1](#) has no effect. See section [5.10.4](#).

1 = Arm the phase adjustment, self clearing

Bit 1: Phase Adjustment Trigger Invert (TINV). This bit specifies the polarity of the trigger signal. See section [5.10.4](#).

0 = Trigger signal normal polarity

1 = Trigger signal inverted

Bit 0: Phase Adjust/Alignment Mode (MODE). This field sets the mode of the phase change. In output phase *alignment* mode, the device resets the MSDIV and LSDIV dividers for all participating outputs so that they are all aligned and then adjusts the phase of each participating output as specified in the [OCxPH](#) register. In output phase *adjustment* mode the device does not reset the MSDIV and LSDIV dividers and therefore causes each participating output to have the phase adjustment specified in the [OCxPH](#) register relative to that output's previous phase. See section [5.10.4](#).

0 = Phase alignment mode

1 = Phase adjustment mode

Register Name: PACR2
Register Description: Phase Adjust Configuration Register 2
Register Address: 1Ch

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	ARMSRC[3:0]					TRGSRC[3:0]		
Default	0	0	0	0	0	0	0	0

Block A for top-level configurations 1 and 2. Block B for top-level configuration 3.

Bits 7 to 4: Output Phase Adjustment Arm Source (ARMSRC[3:0]). This field selects the source of the phase adjustment arming signal. See section [5.10.4](#).

- 0000 = {do not use}
- 0001 = [PACR1](#).ARM bit (one-shot)
- 0010 = APLL transition from unlocked to locked
- 0011 = DPLL transition from unlocked to locked
- 0100 to 0111 = {unused values}
- 1000 = GPIO0 transition (see note below)
- 1001 = GPIO1 transition
- 1010 = GPIO2 transition
- 1011 = GPIO3 transition
- 1100 to 1111 = {unused values}

Bits 3 to 0: Output Phase Adjustment Trigger Source (TRGSRC[3:0]). This field selects the source of the phase adjustment trigger signal. The phase adjustment must be armed or the trigger signal is ignored. The trigger source transition initiates the phase adjustment event. See section [5.10.4](#).

- 0000 = [PACR1](#).TRIG bit
- 0001 = APLL transition from unlocked to locked
- 0010 = DPLL transition from unlocked to locked
- 0011 = DPLL input clock rising edge (after PFD divider)
- 0100 to 0111 = {unused values}
- 1000 = GPIO0 transition (see note below)
- 1001 = GPIO1 transition
- 1010 = GPIO2 transition
- 1011 = GPIO3 transition
- 1100 to 1111 = {unused values}

Note: In both fields above the GPIO transitions are 0-to-1 when [GPIOCR1](#).GPIOxC=0000 and 1-to-0 when [GPIOCR1](#).GPIOxC=0001.

Register Name: MABCR1
Register Description: Measure A-to-B Configuration Register 1
Register Address: 1Dh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RST	START	—	—	—			
Default	0	0	0	0	0	0	0	0

Block A for top-level configurations 1 and 2. Block B for top-level configuration 3.

Bit 7: Measurement Reset (RST). This field stops the current A-to-B phase measurement. This bit is self clearing. See section [5.10.6](#).

1 = Stop measurement (self-clearing)

Bit 6: Measurement Start (START). This field starts a new A-to-B phase measurement. This bit is self clearing. See section [5.10.6](#).

1 = Start new measurement (self-clearing)

Bits 2 to 0: Measurement Time Base Source (TBSRC[2:0]). This field selects the source of the measurement time base. See section [5.10.6](#).

000 = External oscillator

(This is the signal on XA (if an external XO is connected to XA) or the crystal frequency doubled (if a crystal is connected to XA and XB.)

001 = Medium-speed divider 1 (MSDIV1) output clock

010 = Medium-speed divider 2 (MSDIV2) output clock

011 = Medium-speed divider 3 (MSDIV3) output clock

Register Name: MABCR2
Register Description: Measure A-to-B Configuration Register 2
Register Address: 1Eh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	AINV	—	—			ASRC[4:0]		
Default	0	0	0	0	0	0	0	0

Block A for top-level configurations 1 and 2. Block B for top-level configuration 3.

Bit 7: Measurement Input A Invert (AINV). This field inverts the signal of measurement input A. See section [5.10.6](#).

0 = Measure to and from rising edge of input A

1 = Measure to and from falling edge of input A

Bits 4 to 0: Measurement Input A Source (ASRC[4:0]). This field selects the source of measurement input A. See section [5.10.6](#).

00000 = IC1

00001 = IC2

00010 = IC3

01000 = GPIO0

01001 = GPIO1

01010 = GPIO2

01011 = GPIO3

10000 = OC1

10001 = OC2

10010 = OC3

Register Name: MABCR3
Register Description: Measure A to B Configuration Register 3
Register Address: 1Fh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	BINV	—	—			BSRC[4:0]		
Default	0	0	0	0	0	0	0	0

Block A for top-level configurations 1 and 2. Block B for top-level configuration 3.

Bit 7: Measurement Input B Invert (BINV). This field inverts the signal of measurement input B. See section [5.10.6](#).

- 0 = Measure to and from rising edge of input B
- 1 = Measure to and from falling edge of input B

Bits 4 to 0: Measurement Input B Source (BSRC[4:0]). This field selects the source of measurement input B. See section [5.10.6](#).

- 00000 = IC1
- 00001 = IC2
- 00010 = IC3
- 01000 = GPIO0
- 01001 = GPIO1
- 01010 = GPIO2
- 01011 = GPIO3
- 10000 = OC1
- 10001 = OC2
- 10010 = OC3

6.3.2 Status Registers

Register Name: ID1
Register Description: Device Identification Register, MSB
Register Address: 30h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name				<u>IDU[7:0]</u>				
Default	0	0	0	1	1	0	0	1

Bits 7 to 0: Device ID Upper (IDU[7:0]). This field is the upper eight bits of the device ID. Both macro-blocks (A and B) have the same value.

Register Name: ID2
Register Description: Device Identification Register, LSB and Revision
Register Address: 31h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name				<u>IDL[3:0]</u>		<u>REV[3:0]</u>		
Default	1	0	0	1	0	0	0	1

Bits 7 to 4: Device ID Lower (IDL[3:0]). This field is the lower four bits of the device ID.

Bits 3 to 0: Device Revision (REV[3:0]). These bits are the device hardware revision starting at 0.

Register Name: CFGSR
Register Description: Configuration Status Register
Register Address: 40h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TEST	XOFAIL	—	—	IF[1:0]	—	—	AC[1:0]
Default	see below	0	0	0	see below	see below	see below	see below

Bit 7: Test Mode (TEST). This read-only bit is the latched state of the TEST/GPIO2 pin when the RSTN pin transitions high. For proper operation it should be 0. See section [5.4](#).

Bit 6: XO Fail (XOFAIL). This read-only bit is set when the external oscillator signal on the XA pin fails or when the crystal connected to the XA/XB pins fails to oscillate. When the ring oscillator is disabled ([MCR1.ROSCD=1](#)) this bit is always 0.

Bits 3 to 2: Interface Mode (IF[1:0]). These read-only bits are the latched state of the IF1/MISO and IF0/CSN pins when the RSTN pin transitions high. See section [5.4](#).

Bits 1 to 0: Auto-Configuration (AC[1:0]). These read-only bits are the latched state of the AC1/GPIO1 and AC0/GPIO0 pins when the RSTN pin transitions high. See section [5.4](#).

Register Name: GPIOSR
Register Description: GPIO Status Register
Register Address: 41h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	GPIO3	GPIO2	GPIO1	GPIO0
Default	0	0	0	0	0	0	0	0

Bit 3: GPIO3 State (GPIO3). This real-time status bit indicates the current state of the GPIO3 pin, not influenced by any inversion that may be specified by [GPIOCR2.GPIO3C](#).

0 = low

1 = high

Bit 2: GPIO2 State (GPIO2). This real-time status bit indicates the current state of the GPIO2 pin, not influenced by inversion that may be specified by [GPIOCR2.GPIO2C](#).

0 = low

1 = high

Bit 1: GPIO1 State (GPIO1). This real-time status bit indicates the current state of the GPIO1 pin, not influenced by inversion that may be specified by [GPIOCR1.GPIO1C](#).

0 = low

1 = high

Bit 0: GPIO0 State (GPIO0). This real-time status bit indicates the current state of the GPIO0 pin, not influenced by inversion that may be specified by [GPIOCR1.GPIO0C](#).

0 = low

1 = high

Register Name: INTSR
Register Description: Interrupt Status Register
Register Address: 42h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	<u>GLOB</u>	<u>OC</u>	<u>IC</u>	<u>DPLL</u>	<u>APLL</u>	<u>INTIE</u>	<u>INT</u>
Default	0	0	0	0	0	0	0	0

Bit 6: Global Interrupt Status (GLOB). This read-only bit is set if any of the global interrupt status bits are set in the [GLOBISR](#) register. See section [5.12](#).

Bit 5: Output Clock Interrupt Status (OC). This read-only bit is set if any of the output clock interrupt status bits are set in the [OCISR](#) register. See section [5.12](#).

Bit 4: Input Clock Interrupt Status (IC). This read-only bit is set if any of the input clock interrupt status bits are set in the [ICISR](#) register. See section [5.12](#).

Bit 3: DPLL Interrupt Status (DPLL). This read-only bit is set if any of the DPLL interrupt status bits are set in the [DPLLISR](#) register. See section [5.12](#).

Bit 2: APLL Interrupt Status (APLL). This read-only bit is set if any of the APLL interrupt status bits are set in the [APLLISR](#) register. See section [5.12](#).

Bit 1: Interrupt Enable Bit (INTIE). This is the global interrupt enable bit. When this bit is 0 all interrupt sources are prevented from setting the INT global interrupt status bit (below). See section [5.12](#).

0 = Interrupts are disabled at the global level

1 = Interrupts are enabled at the global level

Bit 0: Interrupt Status (INT). This read-only bit is set when any of the GLOB, OC, IC, DPLL or APLL bits in this [INTSR](#) register are set and the INTIE bit is set. This bit can cause an interrupt request when set by configuring one of the GPIO pins to follow it. See section [5.12](#).

0 = No interrupt

1 = An unmasked interrupt source is active

Register Name: GLOBISR
Register Description: Global Functions Interrupt Status Register
Register Address: 43h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	<u>BCDONE</u>	—	—	—	—	—	<u>PA</u>	<u>MAB</u>
Default	see below	0	0	0	0	0	0	0

Bit 7: Boot Controller Done (BCDONE). This bit indicates the status of the on-chip boot controller, which performs auto-configuration from EEPROM. It is cleared when the device is reset and set after the boot controller finishes auto-configuration. See section [5.15](#).

Bit 1: Phase Adjust Interrupt Status (PA). This bit indicates the current status of the interrupt sources from the phase adjust function (see section [5.10.4](#)). It is set when any latched status bit in the [PASR](#) register is set and the associated interrupt enable bit is also set. See section [5.12](#).

Bit 0: Measure AB Interrupt Status (MAB). This bit indicates the current status of the interrupt sources from the A-to-B phase offset measurement function (see section [5.10.6](#)). It is set when any latched status bit in the [MABSR1](#) register is set and the associated interrupt enable bit is also set. See section [5.12](#).

Register Name: ICISR
Register Description: Input Clock Interrupt Status Register
Register Address: 44h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	IC3	IC2	IC1
Default	0	0	0	0	0	0	0	0

Bit 2: Input Clock 3 Interrupt Status (IC3). This bit indicates the current status of the interrupt sources for IC3. This bit is set when latched status [VALSR2](#).IC3L is set and the associated interrupt enable bit is also set. See section [5.12](#).

Bit 1: Input Clock 2 Interrupt Status (IC2). This bit indicates the current status of the interrupt sources for IC2. This bit is set when latched status [VALSR1](#).IC2L is set and the associated interrupt enable bit is also set. See section [5.12](#).

Bit 0: Input Clock 1 Interrupt Status (IC1). This bit indicates the current status of the interrupt sources for IC1. This bit is set when latched status [VALSR1](#).IC1L is set and the associated interrupt enable bit is also set. See section [5.12](#).

Register Name: OCISR
Register Description: Output Clock Interrupt Status Register
Register Address: 45h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	OC3	OC2	OC1
Default	0	0	0	0	0	0	0	0

Bit 2: Output Clock 3 Interrupt Status (OC3). This bit indicates the current status of the interrupt sources for OC3. It is set when any latched status bit in the [OC3SR](#) register is set and the associated interrupt enable bit is also set. See section [5.12](#).

Bit 1: Output Clock 2 Interrupt Status (OC2). This bit indicates the current status of the interrupt sources for OC2. It is set when any latched status bit in the [OC2SR](#) register is set and the associated interrupt enable bit is also set. See section [5.12](#).

Bit 0: Output Clock 1 Interrupt Status (OC1). This bit indicates the current status of the interrupt sources for OC1. It is set when any latched status bit in the [OC1SR](#) register is set and the associated interrupt enable bit is also set. See section [5.12](#).

Register Name: APLLISR
Register Description: APLL Interrupt Status Register
Register Address: 46h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	—	—	APLL
Default	0	0	0	0	0	0	0	0

Bit 0: APLL Interrupt Status (APLL). This bit indicates the current status of the interrupt sources for the APLL. It is set when any latched status bit in the [APLLSR](#) register is set and the associated interrupt enable bit is also set. See section [5.12](#).

Register Name: DPLLISR
Register Description: DPLL Interrupt Status Register
Register Address: 47h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	—	—	<u>DPLL</u>
Default	0	0	0	0	0	0	0	0

Bit 0: DPLL Interrupt Status (DPLL). This bit indicates the current status of the interrupt sources for the DPLL. It is set when any latched status bit in the [DSRL1](#) or [DSRL2](#) register is set and the associated interrupt enable bit in the [DSRIE1](#) or [DSRIE2](#) register is also set. See section [5.12](#).

Register Name: APULLSR
Register Description: APULL Status Register
Register Address: 48h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	ALK2IE	ALK2L	ALK2	ADLK	ALKIE	ALKL	ALK
Default	0	0	0	0	0	0	0	0

Bit 6: APULL Lock 2 Interrupt Enable (ALK2IE). This bit enables the ALK2L latched status bit to send an interrupt request into the interrupt logic.

0 = Interrupt is disabled
 1 = Interrupt is enabled

Bit 5: APULL Lock 2 Latched Status (ALK2L). This latched status bit is set to 1 when the ALK2 status bit changes state (set or cleared). ALK2L is cleared when written with a 1. When ALK2L is set it can cause an interrupt request if the ALK2IE interrupt enable bit is set.

Bit 4: APULL Lock Status 2 (ALK2). This real-time status bit provides one type of APULL lock status. System software should consider the APULL locked when ALK (bit 0) is set to 1 AND ALK2=1. See section [5.8.7](#).

Bit 3: Both APULL and DPLL Locked (ADLK). This real-time status bit indicates when both the APULL and the DPLL are locked.

1 = APULL is locked and DPLL is locked

Bit 2: APULL Lock Interrupt Enable (ALKIE). This bit enables the ALKL latched status bit to send an interrupt request into the interrupt logic.

0 = Interrupt is disabled
 1 = Interrupt is enabled

Bit 1: APULL Lock Latched Status (ALKL). This latched status bit is set to 1 when the ALK status bit changes state (set or cleared). ALKL is cleared when written with a 1. When ALKL is set it can cause an interrupt request if the ALKIE interrupt enable bit is set.

Bit 0: APULL Lock Status (ALK). This real-time status bit indicates one type of APULL lock status. System software should consider the APULL locked when ALK=1 AND ALK2 (bit 4) is set to 1. See section [5.8.7](#).

0 = Not locked
 1 = Locked

Register Name: PTAB1
Register Description: Priority Table Register 1
Register Address: 49h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	REF1[1:0]	—	—	—	SELREF[1:0]	
Default	0	0	0	0	0	0	0	0

Block A for all top-level configurations. Block B for top-level configurations 2 and 3 only.

Bits 5 to 4: Highest Priority Valid Reference (REF1[1:0]). This real-time status field indicates the DPLL's highest-priority valid input reference. Note that an input reference cannot be indicated in this field if it has been marked invalid in the [VALCR1](#) register. When input switching is nonrevertive ([ICSCR1.REVERT=0](#)) this field may not have the same value as the SELREF field. See section [5.7.3.2](#).

- 00 = No valid input reference available
- 01 = IC1 input
- 10 = IC2 input
- 11 = IC3 input

Bits 1 to 0: Selected Reference (SELREF[1:0]). This real-time status field indicates the DPLL's current selected reference. Note that an input clock cannot be indicated in this field if it has been marked invalid in the [VALCR1](#). When input switching is nonrevertive ([ICSCR1.REVERT=0](#)) this field may not have the same value as the REF1[3:0] field. See section [5.7.3.2](#).

- 00 = No valid input reference available
- 01 = IC1 input
- 10 = IC2 input
- 11 = IC3 input

Register Name: PTAB2
Register Description: Priority Table Register 2
Register Address: 4Ah

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	REF3[1:0]	—	—	—	REF2[1:0]	
Default	0	0	0	0	0	0	0	0

Block A for all top-level configurations. Block B for top-level configurations 2 and 3 only.

Bits 5 to 4: Third Highest Priority Valid Reference (REF3[1:0]). This real-time status field indicates the DPLL's third highest-priority valid input reference. Note that an input reference cannot be indicated in this field if it has been marked invalid in the [VALCR1](#) register. See section [5.7.3.2](#).

- 00 = No valid input reference available
- 01 = IC1 input
- 10 = IC2 input
- 11 = IC3 input

Bits 1 to 0: Second Highest Priority Valid Reference (REF2[1:0]). This real-time status field indicates the DPLL's second highest priority validated input reference. Note that an input reference cannot be indicated in this field if it has been marked invalid in the [VALCR1](#) register. See section [5.7.3.2](#).

- 00 = No valid input reference available
- 01 = IC1 input
- 10 = IC2 input
- 11 = IC3 input

Register Name: MABSR1
Register Description: Measure A to B Status Register 1
Register Address: 4Bh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	OVFL	RDYIE	RDYL	<u>BUSY</u>		MEAS[10:8]	
Default	0	0	0	0	0	0	0	0

Block A for top-level configurations 1 and 2. Block B for top-level configuration 3.

Bit 6: Measurement Overflow (OVFL). This latched status bit is set when the phase measurement is ready and an overflow has occurred. See section [5.10.6](#).

0 = No measurement overflow, MEAS is a valid value
 1 = A measurement overflow occurred, MEAS is not a valid value

Bit 5: Measurement Ready Interrupt Enable (RDYIE). This bit enables the RDYL latched status bit to send an interrupt request into the interrupt logic.

0 = Interrupt is disabled
 1 = Interrupt is enabled

Bit 4: Measurement Ready (RDYL). This latched status bit is set when a new phase measurement is ready. See section [5.10.6](#).

0 = New measurement not ready
 1 = A new measurement in MEAS is ready

Bit 3: Measurement Busy (BUSY). This real-time status bit is set when a new phase measurement is being performed. See section [5.10.6](#).

0 = A measurement is not being performed
 1 = A measurement is being performed

Bits 2 to 0: Measurement Value (MEAS[10:8]). See the [MABSR2](#) register description.

Register Name: MABSR2
Register Description: Measure A to B Status Register 2
Register Address: 4Ch

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					MEAS[7:0]			
Default	0	0	0	0	0	0	0	0

Block A for top-level configurations 1 and 2. Block B for top-level configuration 3.

Bits 7 to 0: Measurement Value (MEAS[7:0]). The full 11-bit MEAS[10:0] field spans this register and the lower bits of [MABSR1](#). The format is 2's-complement. This field indicates the result of the A-to-B measurement when BUSY=0, RDYL=1 and OVFL=0. Its value is in units of the selected time-base period and has a range of +1023 to -1024 periods. See section [5.10.6](#).

Register Name: PASR
Register Description: Phase Adjust Status Register
Register Address: 4Dh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	ADJIE	ADJL	<u>BUSY</u>	<u>ARMED</u>
Default	1	0	0	0	0	1	0	0

Block A for top-level configurations 1 and 2. Block B for top-level configuration 3.

Bit 3: Phase Adjustment Finished Interrupt Enable (ADJIE). This bit enables the ADJL latched status bit to send an interrupt request into the interrupt logic.

0 = Interrupt is disabled

1 = Interrupt is enabled

Bit 2: Phase Adjustment Finished (ADJL). This latched status bit is set when the output phase adjustment is completed for all participating outputs. Writing a 1 to this bit clears it. See section [5.10.4](#).

0 = Output phase adjustment has not completed

1 = Output phase adjustment has completed

Note: For phase adjustment when any OCx output's MSDIV period is less than master clock period, this bit may or may not be set depending on exact device configuration and therefore should not be checked by system software. Instead the state of the BUSY real-time bit (bit 1) should be checked.

Bit 1: Phase Adjustment Busy (BUSY). This bit is a real time status that indicates that the output phase adjustment has been triggered and is in progress on the participating outputs. See section [5.10.4](#).

0 = Output phase adjustment is not in progress

1 = Output phase adjustment is in progress

Bit 0: Phase Adjustment Armed (ARMED). This bit is a real time status that indicates that the output phase adjustment is armed and waiting for a trigger. It is cleared when the trigger event occurs. See section [5.10.4](#).

0 = Output phase adjustment is not armed

1 = Output phase adjustment is armed

Register Name: VALSR1
Register Description: Input Clock Valid Status Register 1
Register Address: 4Eh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	IC2IE	IC2L	IC2	—	IC1IE	IC1L	IC1
Default	0	0	1	0	0	0	1	0

Block A for all top-level configurations. Block B for top-level configurations 2 and 3 only.

Bit 6: Input Clock 2 Valid Interrupt Enable (IC2IE). This bit enables the [VALSR1](#).IC2L latched status bit to send an interrupt request into the interrupt logic.

0 = Interrupt is disabled

1 = Interrupt is enabled

Bit 5: Input Clock 2 Valid Latched Status (IC2L). This latched status bit is set to 1 when the [VALSR1](#).IC2 status bit changes state (set or cleared). This bit is cleared when written with a 1 and not set again until the [VALSR1](#).IC2 status bit changes state again. This bit can be the source of an interrupt request. See section [5.7.2](#).

0 = IC2 bit has not changed state since last cleared

1 = IC2 bit has changed state since last cleared

Bit 4: Input Clock 2 Valid (IC2). This real-time status bit is set to 1 when IC2 is valid ([ICxSR.VALT=1](#) and [ICxSR.LKTO=0](#)). See section [5.7.2](#).

0 = Invalid

1 = Valid

Bit 2: Input Clock 1 Valid Interrupt Enable (IC1IE). This bit enables the [VALSR1.IC1L](#) latched status bit to send an interrupt request into the interrupt logic.

0 = Interrupt is disabled

1 = Interrupt is enabled

Bit 1: Input Clock 1 Valid Latched (IC1L). This latched status bit is set to 1 when the [VALSR1.IC1](#) status bit changes state (set or cleared). This bit is cleared when written with a 1 and not set again until the [VALSR1.IC1](#) bit changes state again. This bit can be the source of an interrupt request. See section [5.7.2](#).

0 = IC1 bit has not changed state since last cleared

1 = IC1 bit has changed state since last cleared

Bit 0: Input Clock 1 Valid (IC1). This real-time status bit is set to 1 when IC1 is valid ([ICxSR.VALT=1](#) and [ICxSR.LKTO=0](#)). See section [5.7.2](#).

0 = Invalid

1 = Valid

Register Name: VALSR2
Register Description: Input Clock Valid Status Register 2
Register Address: 4Fh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	IC3IE	IC3L	IC3
Default	0	0	0	0	0	0	1	0

Block A for all top-level configurations. Block B for top-level configurations 2 and 3 only.

Bit 2: Input Clock 3 Valid Interrupt Enable (IC3IE). This bit enables the [VALSR2.IC3L](#) latched status bit to send an interrupt request into the interrupt logic.

0 = Interrupt is disabled

1 = Interrupt is enabled

Bit 1: Input Clock 3 Valid Latched (IC3L). This latched status bit is set to 1 when the [VALSR2.IC3](#) status bit changes state (set or cleared). This bit is cleared when written with a 1 and not set again until the [VALSR2.IC3](#) bit changes state again. This bit can be the source of an interrupt request. See section [5.7.2](#).

0 = IC3 bit has not changed state since last cleared

1 = IC3 bit has changed state since last cleared

Bit 0: Input Clock 3 Valid (IC3). This real-time status bit is set to 1 when IC3 is valid ([ICxSR.VALT=1](#) and [ICxSR.LKTO=0](#)). See section [5.7.2](#).

0 = Invalid

1 = Valid

Register Name: ICxSR
Register Description: Input Clock Status Register
Register Address: IC1: 50h, IC2: 51h, IC3: 52h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	LKCLR	LKTO	LOS	ACVAL	PCVAL	PPVAL	VAL	VALT
Default	0	0	0	1	1	1	0	0

Block A for all top-level configurations. Block B for top-level configurations 2 and 3 only.

Bit 7: Input Clock Lock Timeout Clear (LKCLR). This bit clears the LKTO bit when a 1 is written to it. LKCLR is self-clearing.

1 = Clear the LKTO status bit

Bit 6: Input Clock Lock Timeout (LKTO). This bit is set when ICx is the selected reference to the DPLL and the DPLL has not locked within the lock time-out duration. When LKTO=1 ICx is invalidated by forcing [VALSR](#).ICx low. LKTO can be cleared by writing the LKCLR bit or automatically after a configurable period of time. See section [5.7.2](#).

0 = The input clock lock timeout has not occurred

1 = The input clock lock timeout has occurred

Bit 5: Input Clock Loss Of Signal (LOS). This real-time status bit follows the value of the LOS source selected by [MONxCR2](#).LOSS. See section [5.7.2.1](#).

0 = No loss of signal indicated from an external source

1 = Loss of signal indicated from an external source

Bit 4: Input Clock Activity Monitor Valid (ACVAL). This real-time status bit is high when the activity monitor indicates the input clock is valid and LOS=0. See section [5.7.2](#).

0 = Not valid

1 = Valid

Bit 3: Input Clock Percent Frequency Offset Monitor Valid (PCVAL). This real-time status bit is high when the percent frequency monitor indicates the input clock is valid and LOS=0 and ACVAL=1. See section [5.7.2](#).

0 = Not valid

1 = Valid

Bit 2: Input Clock PPM Frequency Offset Monitor Valid (PPVAL). This real-time status bit is high when the PPM frequency offset monitor indicates the input clock is valid and LOS=0 and ACVAL=1 and PCVAL=1. See section [5.7.2](#).

0 = Not valid

1 = Valid

Bit 1: Input Clock Valid (VAL). This real-time status bit is high when all of the input status bits indicate the input clock is valid (LKTO=0, LOS=0, ACVAL=1, PCVAL=1 and PPVAL=1). It is low when any of the status bits indicate the input clock is not valid. See section [5.7.2.2](#).

0 = The input clock is not valid

1 = The input clock is valid

Bit 0: Input Clock Valid Timed (VALT). This bit is set when the VAL bit has been set for the duration of the validation timer. It is cleared when the VAL bit clears. See section [5.7.2.2](#).

0 = The input clock has not been valid for the specified duration

1 = The input clock has been valid for the specified duration

Register Name: OCxSR
Register Description: Output Clock x Status Register
Register Address: OC1: 53h, OC2: 54h, OC3: 55h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	LSCLKIE	LSCLKL	LSCLK	STARTIE	STARTL	STOPIE	STOPL	STOPD
Default	0	0	0	0	see note	0	see note	see note

In top-level configurations 1 and 2 (section 5.3.1) the output logic and registers are in Block A. (The Block B OC2P_B output is automatically configured by the evaluation software.) In top-level configuration 3 (section 5.3.2) the output logic and registers are in Block B. (The Block A OC2P_A and AC0/GPIO0_A outputs are automatically configured by the evaluation software.)

Bit 7: (LSCLKIE). This bit enables the LSCLKL latched status bit to send an interrupt request into the interrupt logic.

- 0 = Interrupt is disabled
- 1 = Interrupt is enabled

Bit 6: (LSCLKL). This latched status bit is set when the low-speed divider output clock transitions low-to-high. Writing a 1 to this bit clears it.

- 0 = Low speed output clock has not transitioned low to high
- 1 = Low speed output clock has transitioned low to high

Bit 5: (LSCLK). This real-time status bit follows the level of the low-speed divider output clock when the OCxCR3.SRLSEN bit is set.

- 0 = LSCLK is high
- 1 = LSCLK is low

Bit 4: (STARTIE). This bit enables the STARTL latched status bit to send an interrupt request into the interrupt logic.

- 0 = Interrupt is disabled
- 1 = Interrupt is enabled

Bit 3: (STARTL). This latched status bit is set when the output clock signal has been started after being stopped. Writing a 1 to this bit clears it. See section 5.10.5.

- 0 = Output clock signal has not resumed from being stopped
- 1 = Output clock signal has resumed from being stopped

Bit 2: (STOPIE). This bit enables the STOPL latched status bit to send an interrupt request into the interrupt logic.

- 0 = Interrupt is disabled
- 1 = Interrupt is enabled

Bit 1: (STOPL). This latched status bit is set when the output clock signal has been stopped. Writing a 1 to this bit clears it. See section 5.10.5.

- 0 = Output clock signal has not stopped
- 1 = Output clock signal has stopped

Bit 0: (STOPD). This real-time status bit is high when the output clock signal is stopped and low when the output clock is not stopped. See section 5.10.5.

- 0 = Output clock signal is not stopped
- 1 = Output clock signal is stopped

Note: STARTL, STOPL and STOPD are controlled by logic that does not have a clock at reset. Therefore their reset values are indeterminate. They will become 0 when the output clock path is configured and one of the high-speed clocks from the APLL is connected to the logic.

Register Name: DSRR1
Register Description: DPLL Status Register Real-Time 1
Register Address: 56h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	LOL	TRK	HO	FLOCK	—	—
Default	0	0	0	0	0	0	0	0

Block A for all top-level configurations. Block B for top-level configurations 2 and 3 only.

Bit 5: DPLL Loss of Lock (LOL). When the TRK bit is 1, this real-time status bit indicates whether the DPLL is locked to the selected reference or not locked (i.e. “loss of lock”). When TRK=0 this LOL bit has no meaning. See section [5.8.2](#).

0 = Locked
 1 = Loss of lock

Bit 4: DPLL Tracking State (TRK). This real-time status bit indicates when the DPLL is in the tracking state. See section [5.8.2](#).

0 = Not tracking (see HO bit for additional state)
 1 = Tracking (see LOL bit for additional state)

Bit 3: DPLL Holdover State (HO). When the TRK bit is 0, this real-time status bit indicates whether the DPLL is in the freerun state or the holdover state. When TRK=1 this HO bit has no meaning. See section [5.8.2](#).

0 = Freerun
 1 = Holdover

Bit 2: DPLL Fast Lock (FLOCK). When the TRK bit is 1 this real-time status bit indicates that the DPLL is tracking using its fast-lock behavior. When TRK=0 this FLOCK bit has no meaning. See section [5.8.2](#).

Register Name: DSRR2
Register Description: DPLL Status Register Real-Time 2
Register Address: 57h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	NORUN	—	FLIM	—	PSLIM
Default	0	0	0	0	0	0	0	0

Bit 4: DPLL Won't Run (NORUN). (Block A for all top-level configurations. Block B for top-level configurations 2 and 3 only.) The DPLL sets this bit to indicate that it refuses to run because the DSP code loaded into it is for another Microsemi part number. The DPLL stays in free-run mode until the block is reset and loaded with ZL30621 DSP code.

Bit 2: DPLL Frequency Limited (FLIM). (Block A only) This real-time status bit indicates when the DPLL is operating at its configured frequency offset limit. See section [5.8.3](#).

Bit 0: DPLL Phase-Slope Limiting Active (PSLIM). (Block A only) This real-time status bit indicates when the DPLL is actively limiting its phase rate-of-change per its configured limit. See section [5.8.3](#).

Register Name: DSRL1
Register Description: DPLL Status Register Latched 1
Register Address: 59h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	LOLL	TRKL	HOL	FLOCKL	—	—
Default	0	0	0	0	0	0	0	0

Block A for all top-level configurations. Block B for top-level configurations 2 and 3 only.

Bit 5: DPLL Loss of Lock Latched Status (LOLL). This latched status bit is set when [DSRR1.LOL](#) bit changes state. Writing a 1 to this bit clears it.

0 = LOL has not changed state
 1 = LOL has changed state

Bit 4: DPLL Tracking State Latched Status (TRKL). This latched status bit is set when [DSRR1.TRK](#) bit changes state. Writing a 1 to this bit clears it.

0 = TRK has not changed state
 1 = TRK has changed state

Bit 3: DPLL Holdover State Latched Status (HOL). This latched status bit is set when [DSRR1.HO](#) bit changes state. Writing a 1 to this bit clears it.

0 = HO has not changed state
 1 = HO has changed state

Bit 2: DPLL Fast Lock Latched Status (FLOCKL). This latched status bit is set when [DSRR1.FLOCK](#) bit changes state. Writing a 1 to this bit clears it.

0 = FLOCK has not changed state
 1 = FLOCK has changed state

Register Name: DSRL2
Register Description: DPLL Status Register Latched 2
Register Address: 5Ah

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	FLIML	—	PSLIML
Default	0	0	0	0	0	0	0	0

Block A only.

Bit 2: DPLL Frequency Limited (FLIML). This latched status bit is set when [DSRR2.FLIM](#) bit changes state. Writing a 1 to this bit clears it.

0 = FLIM has not changed state
 1 = FLIM has changed state

Bit 0: DPLL Phase-Slope Limiting Active (PSLIML). This latched status bit is set when [DSRR2.PSLIM](#) bit changes state. Writing a 1 to this bit clears it.

0 = PSLIM has not changed state
 1 = PSLIM has changed state

Register Name: DSRIE1
Register Description: DPLL Status Register Interrupt Enable 1
Register Address: 5Bh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	LOLIE	TRKIE	HOIE	FLOCKIE	—	—
Default	0	0	0	0	0	0	0	0

Block A for all top-level configurations. Block B for top-level configurations 2 and 3 only.

Bit 5: DPLL Loss of Lock Interrupt Enable (LOLIE). This interrupt enable bit enables the latched status bit [DSRL1.LOLL](#) to set the interrupt status bit [DPLLISR.DPLL](#).

Bit 4: DPLL Tracking State Interrupt Enable (TRKIE). This interrupt enable bit enables the latched status bit [DSRL1.TRKL](#) to set the interrupt status bit [DPLLISR.DPLL](#).

Bit 3: DPLL Holdover State Interrupt Enable (HOIE). This interrupt enable bit enables the latched status bit [DSRL1.HOL](#) to set the interrupt status bit [DPLLISR.DPLL](#).

Bit 2: DPLL Fast Lock Interrupt Enable (FLOCKIE). This interrupt enable bit enables the latched status bit [DSRL1.FLOCKL](#) to set the interrupt status bit [DPLLISR.DPLL](#).

Register Name: DSRIE2
Register Description: DPLL Status Register Interrupt Enable 2
Register Address: 5Ch

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	FLIMIE	—	PSLIMIE
Default	0	0	0	0	0	0	0	0

Block A only.

Bit 2: DPLL Frequency Limited Interrupt Enable (FLIMIE). This interrupt enable bit enables the latched status bit [DSRL2.FLIML](#) to set the interrupt status bit [DPLLISR.DPLL](#).

Bit 0: DPLL Phase-Slope Limiting Active Interrupt Enable (PSLIMIE). This interrupt enable bit enables the latched status bit [DSRL2.PSLIML](#) to set the interrupt status bit [DPLLISR.DPLL](#).

Register Name: DFREQ1
Register Description: DPLL Frequency Status Register 1
Register Address: 65h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					FREQ[7:0]			
Default	0	0	0	0	0	0	0	0

Block A for all top-level configurations. Block B for top-level configurations 2 and 3 only.

Bits 7 to 0: DPLL Frequency Status (FREQ[7:0]). The full 32-bit FREQ[31:0] field spans this register and the DFREQ2 to DFREQ4 registers and is a multiregister field (see section 6.1.3). This read-only field is a 2's-complement signed integer that expresses the offset the DPLL adds to FREQZ to get the DPLL's frequency tuning word.

The fractional frequency offset of the DPLL in ppm is $10^6 \times \text{FREQ} / \text{FREQZ}$.

Note that FREQ must be converted from 2's-complement to signed integer for use in the above expression. FREQZ is the 40-bit nominal frequency tuning word value in the DFREQZ registers.

Register Name: DFREQ2
Register Description: DPLL Frequency Status Register 2
Register Address: 66h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					FREQ[15:8]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Frequency Status (FREQ[15:8]). See the DFREQ1 register description.

Register Name: DFREQ3
Register Description: DPLL Frequency Status Register 3
Register Address: 67h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					FREQ[23:16]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Frequency Status (FREQ[23:16]). See the DFREQ1 register description.

Register Name: DFREQ4
Register Description: DPLL Frequency Status Register 4
Register Address: 68h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					FREQ[31:24]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Frequency Status (FREQ[31:24]). See the DFREQ1 register description.

Register Name: DPHASE1
Register Description: DPLL Phase Status Register 1
Register Address: 7Dh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					PHASE[7:0]			
Default	0	0	0	0	0	0	0	0

Block A only.

Bits 7 to 0: DPLL Phase Status (PHASE[7:0]). The full 32-bit PHASE[31:0] field spans this register and the DPHASE2 to DPHASE4 registers and is a multi-register field (see section 6.1.3). PHASE is a 2's-complement signed integer value that indicates the current value of the phase detector (i.e. the phase difference between DPLL output and DPLL input). This field expresses the phase difference in units of 2^{-10} ns. Positive values represent input clock leading feedback clock (to the left on a scope).

Register Name: DPHASE2
Register Description: DPLL Phase Status Register 2
Register Address: 7Eh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					PHASE[15:8]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Phase Status (PHASE[15:8]). See the [DPHASE1](#) register description.

Register Name: DPHASE3
Register Description: DPLL Phase Status Register 3
Register Address: 7Fh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					PHASE[23:16]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Phase Status (PHASE[23:16]). See [DPHASE1](#) register description

Register Name: DPHASE4
Register Description: DPLL Phase Status Register 4
Register Address: 80h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					PHASE[31:24]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Phase Status (PHASE[31:24]). See [DPHASE1](#) register description

6.3.3 APLL Configuration Registers

In top-level configurations 1 and 2 (section 5.3.1) the output APLL and APLL registers are in Block A. (The Block B APLL is automatically configured by the evaluation software.)

In top-level configuration 3 (section 5.3.2) the output APLL and APLL registers are in Block B. (The Block A APLL is automatically configured by the evaluation software.)

Register Name: APLLCR1
Register Description: APLL Configuration Register 1
Register Address: 100h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	ENHS2	BYPHS2	—
Default	0	0	0	0	0	0	0	0

Bit 2: Enable High-Speed Divider 2 (ENHS2). This bit is an enable/disable control for HSDIV2. When HSDIV2 is disabled, power consumption is reduced as shown in Table 7. HSDIV2 is enabled when [PLLLEN](#).[APLLEN](#)=1, ENHS2=1 and [APLLCR1](#).[BYPHS2](#)=0.

0 = Disable
1 = Enable

Bit 1: Bypass APLL and High-Speed Divider 2 (BYPHS2). This bit controls the mux immediately to the right of HSDIV2 in [Figure 1](#). The mux that provides the bypass signal to this mux is controlled by APLLMUX in [APLLCR3](#).

0 = No bypass
1 = Bypass

Register Name: APLLCR2
Register Description: APLL Configuration Register 2
Register Address: 101h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	HSDIV2[3:0]				HSDIV1[3:0]			
Name	0	0	0	0	0	0	0	0
Default								

Bits 7 to 4: APLL High-Speed Divider 2 (HSDIV2[3:0]). This field controls the APLL's high-speed divider 2 (see [Figure 10](#)). See section 5.9.2.

0000 = Divide by 4	1000 = Divide by 8
0001 = Divide by 4.5	1001 = Divide by 9
0010 = Divide by 5	1010 = Divide by 10
0011 = Divide by 5.5	1011 = Divide by 11
0100 = Divide by 6	1100 = Divide by 12
0101 = Divide by 6.5	1101 = Divide by 13
0110 = Divide by 7	1110 = Divide by 14
0111 = Divide by 7.5	1111 = Divide by 15

Bits 3 to 0: APLL High-Speed Divider 1 (HSDIV1[3:0]). This field controls the APLL's high-speed divider 1 (see Figure 10). See section 5.9.2.

0000 = Divide by 4	1000 = Divide by 8
0001 = Divide by 4.5	1001 = Divide by 9
0010 = Divide by 5	1010 = Divide by 10
0011 = Divide by 5.5	1011 = Divide by 11
0100 = Divide by 6	1100 = Divide by 12
0101 = Divide by 6.5	1101 = Divide by 13
0110 = Divide by 7	1110 = Divide by 14
0111 = Divide by 7.5	1111 = Divide by 15

Register Name: APLLCR3

Register Description: APLL Configuration Register 3

Register Address: 102h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	—	APLLMUX[2:0]	
Default	0	0	0	0	0	0	0	0

Bits 2 to 0: APLL Mux Control (APLLMUX[2:0]). This field controls the APLL input mux. When APLLMUX=0xx the APLL input comes directly from an input clock pin (used for Block B in top-level configuration 1; also used for Block A in top-level configuration 3; see section 5.3). The bypass mux mentioned below is a 2:1 mux that replaces the HSDIV2 signal with a signal sent “to the bypass mux”.

000 = Crystal driver circuit if crystal is connected, otherwise XA input
(clock frequency can be 2x if clock doubler enabled by [MCR1.DBL=1](#))

001 = IC1 input to the APLL and to the bypass mux

010 = IC2 input to the APLL and to the bypass mux

011 = IC3 input to the APLL and to the bypass mux

110 = DPLL output to the APLL, XA (optionally doubled) sent to the bypass mux

111 = DPLL output to the APLL, no signal sent to the bypass mux (set [APLLCR1.BYPHS2=0](#))

The following decodes are for applications where the master clock comes from IC1 rather than XA ([MCR1.MCSEL1=1](#)).

100 = DPLL output to the APLL, IC1 sent to the bypass mux

101 = DPLL output to the APLL, no signal sent to the bypass mux (set [APLLCR1.BYPHS2=0](#))

Register Name: APLLCR4
Register Description: APLL Configuration Register 4
Register Address: 103h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DECPH		PDSS[2:0]		INCPH		PISS[2:0]	
Default	0	0	0	0	0	0	0	0

Bit 7: Decrement Phase (DECPH). When PDSS=000, this bit is the APLL phase decrement control signal. See section [5.9.3](#). Decrement moves the signal earlier in time (to the left on a scope).

Bits 6 to 4: Phase Decrement Source Select (PDSS[2:0]). This field specifies the APLL phase decrement control signal. Every low-to-high transition and every high-to-low transition of the signal decrements the APLL's output phase. See section [5.9.3](#).

000 = DECPH bit
001 = GPIO0
010 = GPIO1
011 = GPIO2
100 = GPIO3
101 to 111 = {unused values}

Bit 3: Increment Phase (INCPH). When PISS=000, this bit is the APLL phase increment control signal. See section [5.9.3](#). Increment moves the signal later in time (to the right on a scope).

Bits 2 to 0: Phase Increment Source Select (PISS[2:0]). This field specifies the APLL phase increment control signal. Every low-to-high transition and every high-to-low transition of the signal increments the APLL's output phase. See section [5.9.3](#).

000 = INCPH bit
001 = GPIO0
010 = GPIO1
011 = GPIO2
100 = GPIO3
101 to 111 = {unused values}

Register Name: AFBDIV1
Register Description: APLL Feedback Divider Register 1
Register Address: 106h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name				AFBDIV[7:0]				
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Register (AFBDIV[7:0]). The full 42-bit AFB DIV[41:0] field spans the AFB DIV1 through AFB DIV6 registers. AFB DIV is an unsigned number with 9 integer bits (AFBDIV[41:33]) and up to 33 fractional bits. AFB DIV specifies the fixed-point term of the APLL's fractional feedback divide value. The value AFB DIV=0 is undefined. Unused least significant bits must be written with 0. AFB DIV[41:40] must be set to 00. See section [5.9.2](#).

Register Name: AFBDIV2
Register Description: APLL Feedback Divider Register 2
Register Address: 107h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	AFBDIV[15:8]							
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Register (AFBDIV[15:8]). See the [AFBDIV1](#) register description.

Register Name: AFBDIV3
Register Description: APLL Feedback Divider Register 3
Register Address: 108h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	AFBDIV[23:16]							
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Register (AFBDIV[23:16]). See the [AFBDIV1](#) register description.

Register Name: AFBDIV4
Register Description: APLL Feedback Divider Register 4
Register Address: 109h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	AFBDIV[31:24]							
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Register (AFBDIV[31:24]). See the [AFBDIV1](#) register description.

Register Name: AFBDIV5
Register Description: APLL Feedback Divider Register 5
Register Address: 10Ah

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	AFBDIV[39:32]							
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Register (AFBDIV[39:32]). See the [AFBDIV1](#) register description.

Register Name: AFBDIV6
Register Description: APLL Feedback Divider Register 6
Register Address: 10Bh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	—	AFBDIV[41:40]	
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Register (AFBDIV[41:40]). See the [AFBDIV1](#) register description.

Register Name: AFBDEN1
Register Description: APLL Feedback Divider Denominator Register 1
Register Address: 10Ch

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	AFBDEN[7:0]							
Default	0	0	0	0	0	0	0	1

Bits 7 to 0: APLL Feedback Divider Denominator Register (AFBDEN[7:0]). The full 32-bit AFBDEN[31:0] field spans AFBDEN1 through AFBDEN4 registers. AFBDEN is an unsigned integer that specifies the denominator of the APLL's fractional feedback divide value. The value AFBDEN=0 is undefined. When [AFBBP=0](#), [AFBDEN](#) must be set to 1. See section [5.9.2](#).

Register Name: AFBDEN2
Register Description: APLL Feedback Divider Denominator Register 2
Register Address: 10Dh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	AFBDEN[15:8]							
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Denominator Register (AFBDEN[15:8]). See the [AFBDEN1](#) register description.

Register Name: AFBDEN3
Register Description: APLL Feedback Divider Denominator Register 3
Register Address: 10Eh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	AFBDEN[23:16]							
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Denominator Register (AFBDEN[23:16]). See the [AFBDEN1](#) register description.

Register Name: AFBDEN4
Register Description: APLL Feedback Divider Denominator Register 4
Register Address: 10Fh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	AFBDEN[31:24]							
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Denominator Register (AFBDEN[31:24]). See the [AFBDEN1](#) register description.

Register Name: AFBREM1
Register Description: APLL Feedback Divider Remainder Register 1
Register Address: 110h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					AFBREM[7:0]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Remainder Register (AFBREM[7:0]). The full 32-bit AFBDEN[31:0] field spans AFBREM1 through AFBREM4 registers. AFBREM is an unsigned integer that specifies the remainder of the APLL's fractional feedback divider value. When **AFBBP**=0, **AFBREM** must be set to 0. See section [5.9.2](#).

Register Name: AFBREM2
Register Description: APLL Feedback Divider Remainder Register 2
Register Address: 111h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					AFBREM[15:8]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Remainder Register (AFBREM[15:8]). See the [AFBREM1](#) register description.

Register Name: AFBREM3
Register Description: APLL Feedback Divider Remainder Register 3
Register Address: 112h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					AFBREM[23:16]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Remainder Register (AFBREM[23:16]). See the [AFBREM1](#) register description.

Register Name: AFBREM4
Register Description: APLL Feedback Divider Remainder Register 4
Register Address: 113h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					AFBREM[31:24]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Remainder Register (AFBREM[31:24]). See the [AFBREM1](#) register description.

Register Name: AFBBP
Register Description: APLL Feedback Divider Truncate Bit Position
Register Address: 114h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	AFBBP[7:0]							
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: APLL Feedback Divider Truncate Bit Position (AFBBP[7:0]). This unsigned integer specifies the number of fractional bits that are valid in the [AFBDIV](#) value. There are 33 fractional bits in AFBDIV. The value in this AFBBP field specifies 33 – number_of_valid_AFBDIV_fractional_bits. When AFBBP=0 all 33 AFBDIV fractional bits are valid. When AFBBP=9, the most significant 24 AFBDIV fractional bits are valid and the least significant 9 bits must be set to 0. This register field is only used when the feedback divider value is expressed in the form [AFBDIV + AFBREM / AFBDEN](#). AFBBP values greater than 33 are invalid. When AFBBP=0, [AFBREM](#) must be set to 0 and [AFBDEN](#) must be set to 1. See section [5.9.2](#).

6.3.4 Output Clock Configuration Registers

In top-level configurations 1 and 2 (section [5.3.1](#)) the output logic and registers are in Block A. (The Block B OC2P_B output is automatically configured by the evaluation software.)

In top-level configuration 3 (section [5.3.2](#)) the output logic and registers are in Block B. (The Block A OC2P_A and AC0/GPIO0_A outputs are automatically configured by the evaluation software.)

Register Name: OCxCR1
Register Description: Output Clock x Configuration Register 1
Register Address: OC1: 200h, OC2: 210h, OC3: 220h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
Name	PHEN	MSDIV[6:0]							
Default	0	0	0	0	0	0	0	0	

Bit 7: Phase Adjust Enable (PHEN). This bit enables this output to participate in phase adjustment/alignment. See section [5.10.4](#).

0 = Phase adjustment/alignment disabled for this output

1 = Phase adjustment/alignment enabled for this output

Bits 6 to 0: Medium-Speed Divider Value (MSDIV[6:0]). This field specifies the setting for the output clock's medium-speed divider. The divisor is MSDIV+1. Note that if MSDIV is not set to 0 (bypass) then MSDIV must be set to a value that causes the output clock of the medium-speed divider to be 425MHz or less. When MSDIV=0, the medium-speed divider, phase adjust, low-speed divider, start/stop and output duty cycle adjustment circuits are bypassed and the high-frequency clock signal is directly sent to the output driver. See section [5.10.2](#).

Register Name: OCxCR2
Register Description: Output Clock x Configuration Register 2
Register Address: OC1: 201h, OC2: 211h, OC3: 221h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	ASQUEL	POL	DRIVE[1:0]		STOPDIS	OCSF[2:0]		
Default	0	0	0	0	0	0	0	0

Bit 7: Auto-Squelch Enable (ASQUEL). This bit enables automatic squelching of the output clock whenever the DPLL has no selected reference (PTAB1.SELREF=0). When OCxSTOP.MODE=01 or 10 the output is squelched at the next positive or negative edge using the START/STOP function. When the output is configured as a CML output or a CMOS complementary output is squelched, its OCxN pin is opposite polarity of the OCxP pin.

0 = Auto-squelch disabled

1 = Auto-squelch enabled

Bit 6: Clock Path Polarity (POL). The clock path to the CML, HSTL and CMOS outputs is inverted when this bit is set. This does not invert the LSDIV path to the CMOS OCxN pin if that path is enabled. See section [5.10.1](#).

Bits 5 to 4: CMOS/HSTL Output Drive Strength (DRIVE[1:0]). The CMOS/HSTL output drivers have four equal sections that can be enabled or disabled to achieve four different drive strengths from 1x to 4x. When the output power supply VDDOx is 3.3V or 2.5V, the user should start with 1x and only increase drive strength if the output is highly loaded and signal transition time is unacceptable. When VDDOx is 1.8V or 1.5V the user should start with 4x and only decrease drive strength if the output signal has unacceptable overshoot. See section [5.10.1](#).

00 = 1x

01 = 2x

10 = 3x

11 = 4x

Bit 3: Stop Disable (STOPDIS). This bit causes the output to become disabled (high impedance) while the output clock is stopped. See section [5.10.5](#).

0 = Do not disable the output while stopped

1 = Disable the output while stopped

Bits 2 to 0: Output Clock Signal Format (OCSF[2:0]). Note that OCEN.OCxEN=0 forces the output driver to be high-impedance regardless of the value of the OCSF register field. See section [5.10.1](#).

000 = Disabled (high-impedance, low power mode)

001 = CML, standard swing ($V_{OD} = 800\text{mV}_{P-P}$ typical)

010 = CML, narrow swing ($V_{OD} = 400\text{mV}_{P-P}$ typical)

011 = HSTL (Set OCxCR2.DRIVE=11 (4x) to meet JESD8-6)

100 = Two CMOS: OCxP in phase with OCxN

101 = One CMOS: OCxP high impedance, OCxN enabled

110 = One CMOS: OCxP enabled, OCxN high impedance

111 = Two CMOS: OCxP inverted vs. OCxN

Register Name: OCxCR3
Register Description: Output Clock x Configuration Register 3
Register Address: OC1: 202h, OC2: 212h, OC3: 222h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	SRLSEN	DIVSEL	NEGLSD	LSSEL	—	—	—	LSDIV[24]
Default	0	0	0	0	0	0	0	0

Bit 7 Enable LSDIV Statuses (SRLSEN). This bit enables the [OCxSR.LSCLK](#) real-time status bit and its associated latched status bit [OCxSR.LSCLKL](#).

- 0 = LSCLK status bit is not enabled (low)
- 1 = LSCLK status bit is enabled

Bit 6: High Speed Divider Select (DIVSEL). This bit selects which high-speed divider is the source of the clock for the output. See section [5.10.2](#).

- 0 = HSDIV1
- 1 = HSDIV2

Bit 5: OCxN Low Speed Divider (NEGLSD). This bit selects the source of the clock on the OCxN pin in CMOS mode. See section [5.10.2](#).

- 0 = Same as OCxP
- 1 = Output of the LSDIV divider

Note: NEGLSD should only be set to 1 in two-CMOS mode ([OCxCR2.OCSF](#)=100 or 111) and when [OCxCR2.POL](#)=0.

Bit 4: LSDIV Select (LSSEL). This bit selects the source of the output clock. When the MSDIV divider is selected (LSSEL=0) the LSDIV divider output can be independently selected as the source for the OCxN pin (in CMOS output mode) or monitored by the [OCxSR.LSCLK](#) status bit. This bit is only valid when [OCxCR1.MSDIV](#) > 0. See section [5.10.2](#).

- 0 = The output clock is sourced from the MSDIV divider.
- 1 = The output clock is sourced from the LSDIV divider.

Bit 0: Low-Speed Divider Value (LSDIV[24]). See the [OCxDIV1](#) register description.

Register Name: OCxDIV1
Register Description: Output Clock x Divider Register 1
Register Address: OC1: 203h, OC2: 213h, OC3: 223h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					LSDIV[7:0]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: Low-Speed Divider Value (LSDIV[7:0]). The full 25-bit LSDIV[24:0] field spans this register, OCxDIV2, OCxDIV3, and bit 0 of OCxCR3. LSDIV is an unsigned integer. The frequency of the clock from the medium-speed divider is divided by LSDIV+1. The OCxCR3.LSSEL and NEGLSD bits control when the output of the low-speed divider is present on the OCxP and OCxN output pins. OCxCR1.MSDIV must be > 0 for the low-speed divider to operate. See section 5.10.2.

Register Name: OCxDIV2
Register Description: Output Clock x Divider Register 2
Register Address: OC1: 204h, OC2: 214h, OC3: 224h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					LSDIV[15:8]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: Low-Speed Divider Value (LSDIV[15:8]). See the OCxDIV1 register description.

Register Name: OCxDIV3
Register Description: Output Clock x Divider Register 3
Register Address: OC1: 205h, OC2: 215h, OC3: 225h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					LSDIV[23:16]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: Low-Speed Divider Value (LSDIV[23:16]). See the OCxDIV1 register description.

Register Name: OCxDC
Register Description: Output Clock x Duty Cycle Register
Register Address: OC1: 206h, OC2: 216h, OC3: 226h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					OCDC[7:0]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: Output Clock Duty Cycle (OCDC[7:0]). This field controls the output clock signal duty cycle when MSDIV>0 and LSDIV>1. When OCDC = 0 the output clock is 50%. Otherwise the clock signal is a pulse with a width of OCDC number of MSDIV output clock periods. The range of OCDC can create pulse widths from 1 to 255 MSDIV output clock periods. When OCxCR2.POL=0, the pulse is high and the signal is low the remainder of the cycle. When POL=1, the pulse is low and the signal is high the remainder of the cycle. See section 5.10.3.

Register Name: OCxPH
Register Description: Output Clock x Phase Adjust Register
Register Address: OC1: 207h, OC2: 217h, OC3: 227h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					PHADJ[7:0]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: Phase Adjust Value (PHADJ[7:0]). When [OCxCR1.PHEN=1](#), this field specifies the phase adjustment of the output clock during a phase adjustment event. When [OCxCR1.PHEN=0](#), this field is ignored. The specified phase adjustment occurs once during a phase adjustment event. The format of the field is 2's-complement with the LSB being one half of an HSDIV output clock period. Positive values move the signal later in time (to the right on a scope). See section [5.10.4](#).

00000000 = 0.0 UI

00000001 = +0.5 UI

00000010 = +1.0 UI

00000011 = +1.5 UI

...

01111110 = +63.0 UI

01111111 = +63.5 UI

10000000 = -64.0 UI

10000001 = -63.5 UI

...

11111101 = -1.5 UI

11111110 = -1.0 UI

11111111 = -0.5 UI

Register Name: OCxSTOP
Register Description: Output Clock x Start Stop Register
Register Address: OC1: 208h, OC2: 218h, OC3: 228h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	STOP	—			SRC[3:0]			MODE[1:0]
Default	0	0	0	0	0	0	0	0

Bit 7: Output Clock Stop (STOP). When SRC=0000, this bit is used to stop the output clock high or low. The output stays stopped while this bit is high. See section [5.10.5](#).

0 = Do not stop the output clock

1 = Stop the output clock

Bits 5 to 2: Output Clock Stop Source (SRC[3:0]). This field specifies the source of the stop signal. See section [5.10.5](#).

0000 = STOP bit

0001 = The arming of a phase adjustment (signal stopped when [PASR.ARMED](#) is asserted; signal started when [PASR.ADJL](#) is asserted)

0010 to 0111 = {unused values}

1000 = GPIO0

1001 = GPIO1

1010 = GPIO2

1011 = GPIO3

1100 to 1111 = {unused values}

Bits 1 to 0: Output Clock Stop Mode (MODE[1:0]). This field selects the mode of the start-stop function. See section [5.10.5](#).

00 = Never stop

01 = Stop High: stop after rising edge of output clock, start after falling edge of output clock

10 = Stop Low: stop after falling edge of output clock, start after rising edge of output clock

11 = {unused value}

The following table shows which pin(s) stop high or low as specified above for each output signal format:

Signal Format	OCxCR2.OCSF	Pin that Stops As Specified
CML	001 or 010	OCxP
HSTL	011	OCxN
Two CMOS, OCxP in phase with OCxN	100	OCxP and OCxN
One CMOS, OCxN enabled	101	OCxN
One CMOS, OCxP enabled	110	OCxP
Two CMOS, OCxP inverted vs. OCxN	111	OCxN

Notes:

1. The highest priority condition for an output is when it is stopped and [OCxCR2.STOPDIS](#)=1. When this condition occurs both OCxP and OCxN become high-impedance regardless of the state of the control bits mentioned below.
2. When the output is not stopped or when [OCxCR2.STOPDIS](#)=0, [OCxCR3.NEGLSD](#)=1 causes the OCxN pin to follow the output clock of the low-speed divider uninverted regardless of the signal format, regardless of the state of [OCxCR2.POL](#), and regardless of whether the output is stopped.
3. When the above situations do not apply, [OCxCR2.POL](#)=1 changes Stop High to Stop Low and vice versa.

6.3.5 Input Clock Configuration Registers

These registers are in Block A.

Register Name: ICxCR1
Register Description: Input Clock x Configuration Register 1
Register Address: IC1: 300h, IC2: 320h, IC3: 340h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	POL	—	—	—	—	—	HSDIV[1:0]
Default	0	0	0	0	0	0	0	0

Bit 6: Locking Polarity (POL). This field specifies which input clock signal edge to will lock to. See section [5.7.1](#).

- 0 = Rising edge
- 1 = Falling edge

Bits 1 to 0: Input Clock High-Speed Divider (HSDIV[1:0]). This field specifies the divide value for the input clock high-speed divider. See section [5.7.1](#).

- 00 = Divide by 1
- 01 = Divide by 2
- 10 = Divide by 4
- 11 = Divide by 8

Register Name: MONxCR2
Register Description: Input Clock Monitor x Configuration Register 2
Register Address: IC1: 311h, IC2: 331h, IC3: 351h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	LOSS[2:0]			—	ACEN	PCEN	PPEN
Default	0	0	0	0	0	0	0	0

Bits 6 to 4: Loss-of-Signal Source Select (LOSS[2:0]). This field selects the source of the loss-of-signal indicator. The selected source drives the [ICxSR.LOS](#) bit and invalidates the input clock. See section [5.7.2.1](#).

- 000 = No source. [ICxSR.LOS](#) is set to 0.
- 001 = GPIO0
- 010 = GPIO1
- 011 = GPIO2
- 100 = GPIO3
- 101 to 111 = {unused values}

Bit 2: Activity Monitor Enable (ACEN). This bit enables the input clock activity monitor. See section [5.7.2](#).

- 0 = Disable
- 1 = Enable

Bit 1: Percent Frequency Monitor Enable (PCEN). This bit enables the input clock percent frequency monitor. See section [5.7.2](#).

- 0 = Disable
- 1 = Enable

Bit 0: Parts Per Million Frequency Monitor Enable (PPEN). This bit enables the input clock PPM frequency monitor. See section [5.7.2](#).

- 0 = Disable
- 1 = Enable

6.3.6 DPLL Configuration Registers

The DPLL and its registers are in Block A. The simple Block B DPLL for jitter attenuation is automatically configured by the evaluation software.

Note: When the DPLL is disabled ([PLLEN](#).DPLLLEN=0) all DPLL register fields are ignored by the device and should be ignored by system software.

Register Name: ICSCR1
Register Description: Input Clock Select Configuration Register 1
Register Address: 400h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	EXTSW	REVERT	—	FBSEL[1:0]	—	—	—	—
Default	0	0	0	0	0	0	0	0

Bit 7: External Reference Switching Mode (EXTSW). This bit enables external reference switching mode. See section [5.7.3.4](#).

0 = Normal operation
 1 = External switching mode

Bit 6: Revertive Mode (REVERT). This bit configures the DPLL for revertive or nonrevertive operation. In revertive mode, if an input clock with a higher priority than the selected reference becomes valid, the higher priority reference immediately becomes the selected reference. In nonrevertive mode the higher priority reference does not immediately become the selected reference but does become the highest-priority reference in the priority table (REF1 field in the [PTAB1](#) register). See section [5.7.3.2](#).

0 = Nonrevertive
 1 = Revertive

Bits 4 to 3: DPLL Feedback Select (FBSEL[1:0]). This field specifies the source of the DPLL's feedback clock. External feedback supports zero-delay applications (i.e. zero phase difference between input clock and output clock). When an input is chosen as the DPLL's feedback clock, that input should have its priority set to 0 (unavailable) in the [IPR](#) registers.

00: Internal feedback: Use DPLL DCO output for feedback
 01: External Feedback through IC1
 10: External Feedback through IC2
 11: External Feedback through IC3

Register Name: VALCR1
Register Description: Input Clock Valid Control Register 1
Register Address: 401h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	IC3	IC2	IC1
Default	0	0	0	0	0	1	1	1

Bits 2 to 0: Input Clock Valid Control (IC3, IC2, IC1). These control bits can be used to force input clocks to be considered invalid. If a clock is invalidated by one of these control bits it will not appear in the priority table in the [PTAB1](#) and [PTAB2](#) registers, even if the input monitor indicates the input is valid. These bits are useful when system software needs to force clocks to be "not available for use". Note that setting a VALCR bit low has no effect on the input monitor status bits for that input clock. See section [5.7.3](#).

0 = Force invalid
 1 = Don't force invalid; input monitor determines validity

Register Name: IPR1
Register Description: Input Priority Register 1
Register Address: 402h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	PRI2[1:0]		—	—	PRI1[1:0]	
Default	0	0	1	0	0	0	0	1

Bits 5 to 4: Priority for Input Clock 2 (PRI2[1:0]). This field specifies the priority of IC2. Priority 01 is highest; priority 11 is lowest. See section [5.7.3.1](#).

00 = IC2 unavailable for selection.
 01–11 = IC2 relative priority

Bits 1 to 0: Priority for Input Clock 1 (PRI1[1:0]). This field specifies the priority of IC1. Priority 01 is highest; priority 11 is lowest. See section [5.7.3.1](#).

00 = IC1 unavailable for selection.
 01–11 = IC1 relative priority

Register Name: IPR2
Register Description: Input Priority Register 2
Register Address: 403h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	—	PRI3[3:0]	
Default	0	0	0	0	0	0	1	1

Bits 1 to 0: Priority for Input Clock 3 (PRI3[1:0]). This field specifies the priority of IC3. Priority 01 is highest; priority 11 is lowest. See section [5.7.3.1](#).

00 = IC3 unavailable for selection.
 01–11 = IC3 relative priority

Register Name: PHLKTO
Register Description: DPLL Phase Lock Timeout Register
Register Address: 404h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name		PHLKTO[1:0]			PHLKTO[5:0]			
Default	0	0	1	1	0	0	1	0

Bits 7 to 6: Phase Lock Timeout Multiplier (PHLKTO[1:0]). This field is an unsigned integer that specifies the resolution of the PHLKTO field below. The DSP rate is calculated by the GUI and displayed at the bottom of the GUI's DPLL window.

- 00 = 1 x DSP rate
- 01 = 16 x DSP rate
- 10 = 256 x DSP rate
- 11 = 4096 x DSP rate

Bits 5 to 0: Phase Lock Timeout (PHLKTO[5:0]). This field is an unsigned integer that, together with the PHLKTO field above, specifies the length of time that the DPLL attempts to lock to an input clock before declaring a phase lock alarm (by setting the corresponding [ICxSR.LKTO](#) bit). The timeout period in seconds is PHLKTO[5:0] x 16^(PHLKTO[1:0]) * (1/DSP_RATE). When unable to declare lock, the DPLL remains in the tracking state for the specified time before declaring a phase lock alarm on the selected input. When PHLKTO=0, the timeout is disabled, and the DPLL can remain indefinitely in the tracking state. The DSP rate is calculated by the GUI and displayed at the bottom of the GUI's DPLL window.

Register Name: LKATO
Register Description: DPLL Lock Alarm Timeout Register
Register Address: 405h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name		LKATOM[1:0]			LKATO[5:0]			
Default	0	0	1	1	0	0	1	0

Bits 7 to 6: Lock Alarm Timeout Multiplier (LKATOM[1:0]). This field is an unsigned integer that specifies the resolution of the LKATO field below. The DSP rate is calculated by the GUI and displayed at the bottom of the GUI's DPLL window.

- 00 = 1 x DSP rate
- 01 = 16 x DSP rate
- 10 = 256 x DSP rate
- 11 = 4096 x DSP rate

Bits 5 to 0: Lock Alarm Timeout (LKATO[5:0]). This field is an unsigned integer that, together with the LKATOM field above, specifies the length of time that a phase lock alarm remains active before being automatically deasserted (by clearing the corresponding [ICxSR.LKTO](#) bit). The timeout period in seconds is LKATO[5:0] x 16^(LKATOM[1:0]) * (1/DSP RATE).. When LKATO=0, the timeout is disabled, the phase lock alarm remains active until cleared by software writing a 1 to the [ICxSR.LKCLR](#) bit. The DSP rate is calculated by the GUI and displayed at the bottom of the GUI's DPLL window.

Register Name: DPLLCR1
Register Description: DPLL Configuration Register 1
Register Address: 40Bh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	HOMODE[1:0]			MODE[3:0]		
Default	0	0	0	0	0	0	0	0

Bits 5 to 4: Holdover Mode (HOMODE[1:0]).

- 00: Averaged Holdover – do not reset HO average when new input selected
- 01: Averaged Holdover – reset HO average when new input selected
- 10: Freerun – do not use HO average but rather nominal frequency tuning word value
- 11: {unused value}

Bits 3 to 0: DPLL Mode (MODE[3:0]). This field selects the operational mode of the DPLL.

- 0000: Reset
- 0100: Freerun
- 0101: Holdover
- 0110: Tracking
- all other values are invalid

Register Name: DPLLCR2
Register Description: DPLL Configuration Register 2
Register Address: 40Ch

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	—	HSEN[1:0]	
Default	0	0	0	0	0	0	0	0

Bits 1 to 0: Hitless Switching Enable (HSEN[1:0]). When hitless switching is disabled the DPLL pulls in to the phase of the new input reference. A hitless switch measures the phase difference between previous input and next input and builds that phase difference into the loop so that DPLL output phase does not change during the input reference change. See section 5.8.3.

- 00 = Hitless switching disabled
- 01 = Switch only: hitless behavior only when switching from one input reference to another
- 10 = Switch or Valid: hitless behavior when switching from one input reference to another or when switching back to the same input reference due to short input dropout or input invalidation

Register Name: DFREQZ1
Register Description: DPLL Frequency Tuning Word Register 1
Register Address: 420h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	FREQZ[7:0]							
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Frequency Tuning Word (FREQZ[7:0]). The full 40-bit DFREQZ[39:0] field spans this register and the DFREQZ2 to DFREQZ5 registers and is a multi-register field (see section 6.1.3). This unsigned coefficient is the frequency tuning word value that sets the frequency the DPLL generates during freerun and in NCO mode. It is also the base frequency to which the DPLL adds a dynamic offset during tracking. The nominal value of this field is set by the evaluation software. In NCO mode, system software can change this value by any fraction (less than 1 ppb to multiple ppm) to cause a fractional change in the DPLL's output frequency. See section 5.8.7.

Register Name: DFREQZ2
Register Description: DPLL Frequency Tuning Word Register 2
Register Address: 421h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	FREQZ[15:8]							
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Frequency Tuning Word (FREQZ[15:8]). See the DFREQZ1 register description.

Register Name: DFREQZ3
Register Description: DPLL Frequency Tuning Word Register 3
Register Address: 422h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	FREQZ[23:16]							
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Frequency Tuning Word (FREQZ[23:16]). See the DFREQZ1 register description.

Register Name: DFREQZ4
Register Description: DPLL Frequency Tuning Word Register 4
Register Address: 423h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	FREQZ[31:24]							
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Frequency Tuning Word (FREQZ[31:24]). See the DFREQZ1 register description.

Register Name: DFREQZ5
Register Description: DPLL Frequency Tuning Word Register 5
Register Address: 424h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	FREQZ[39:32]							
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Frequency Tuning Word (FREQZ[39:32]). See the DFREQZ1 register description.

Register Name: DPHOFF1
Register Description: DPLL Phase Offset Register 1
Register Address: 428h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					PHOFF[7:0]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Phase Offset (PHOFF[7:0]). The full 32-bit PHOFF[31:0] field spans this register and the DPHOFF2 to DPHOFF4 registers and is a multi-register field (see section 6.1.3). This 2s-complement coefficient is used to cause a positive or negative phase offset in the output of the DPLL relative to the input. This field expresses the phase offset in units of 2⁻¹⁰ns. Positive values represent input clock leading feedback clock (to the left on a scope)..

Register Name: DPHOFF2
Register Description: DPLL Phase Offset Register 2
Register Address: 429h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					PHOFF[15:8]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Phase Offset (PHOFF[15:8]). See the [DPHOFF1](#) register description.

Register Name: DPHOFF3
Register Description: DPLL Phase Offset Register 3
Register Address: 42Ah

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					PHOFF[23:16]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Phase Offset (PHOFF[23:16]). See the [DPHOFF1](#) register description.

Register Name: DPHOFF4
Register Description: DPLL Phase Offset Register 4
Register Address: 42Bh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name					PHOFF[31:24]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 0: DPLL Phase Offset (PHOFF[31:24]). See the [DPHOFF1](#) register description.

7. Electrical Characteristics

Absolute Maximum Ratings

Parameter	Symbol	Min.	Max.	Units
Supply voltage, nominal 1.8V	VDD18	-0.3	1.98	V
Supply voltage, nominal 3.3V	VDD33	-0.3	3.63	V
Supply voltage, VDDOx (x=1,2,3)	VDDOx	-0.3	3.63	V
Voltage on XA, any ICxP/N, any OCxP/N pin	VANAPIN	-0.3	3.63	V
Voltage on any digital I/O pin	VDIGPIN	-0.3	5.5	V
Storage Temperature Range	T _{ST}	-55	+125	°C

* Exceeding these values may cause permanent damage. Functional operation under these conditions is not implied.

* Voltages are with respect to ground (VSS) unless otherwise stated.

Note 1: The typical values listed in the tables of Section 7 are not production tested.

Note 2: Specifications to -40°C and 85°C are guaranteed by design or characterization and not production tested.

Table 6 - Recommended DC Operating Conditions

Parameter	Symbol	Min.	Typ.	Max.	Units
Supply voltage, nominal 1.8V	VDD18	1.71	1.8	1.89	V
Supply voltage, nominal 3.3V	VDD33	3.135	3.3	3.465	V
Supply voltage, VDDOx (x=1,2,3)	VDDOx	1.425 1.71 2.375 3.135	1.5 1.8 2.5 3.3	1.575 1.89 2.625 3.465	V
Operating temperature	T _A	-40		+85	°C

Table 7 - Electrical Characteristics: Supply Currents

Characteristics	Symbol	Min.	Typ. ¹	Max	Units	Notes
Total power, one input and one nominal-swing CML output enabled	P _{DISS}		1.4		W	Note 4
Total current, all 1.8V supply pins	I _{DD18}		378	468	mA	Note 2, 4
Total current, all 3.3V supply pins	I _{DD33}		300	368	mA	Note 2, 4
3.3V supply current change from enabling or disabling the crystal driver circuit	ΔI _{DD33XTAL}		16		mA	
3.3V supply current change from enabling or disabling high-speed divider 2	ΔI _{DD33HSD}		20		mA	
1.8V supply current from enabling/disabling per-output mux and dividers using OCEN.OCxEN bit	ΔI _{DD18ODIV}		28		mA	
1.8V supply current change from enabling or disabling a CML output, standard swing	ΔI _{DD18CML}		10		mA	
3.3V supply current change from enabling or disabling a CML output, standard swing	ΔI _{DD33CML}		17		mA	
1.8V supply current change from enabling or disabling a CML output, narrow swing	ΔI _{DD18CMLN}		10		mA	
3.3V supply current change from enabling or disabling a CML output, narrow swing	ΔI _{DD33CMLN}		9		mA	
1.8V supply current change from enabling or disabling a pair of single-ended outputs	ΔI _{DD18CMOS}		2		mA	
VDDOx supply current change from enabling or disabling a pair of single-ended outputs	ΔI _{DD33CMOS}		16		mA	Note 3
1.8V supply current change from enabling or disabling an input clock	ΔI _{DD18IN}		13		mA	

Note 1: Typical values measured at 1.80V and 3.30V supply voltages and 25°C ambient temperature.

Note 2: Max I_{DD} measurements made with all blocks enabled, 650MHz signals on IC1 and IC2 inputs, 187.5MHz signal on IC3, VCO frequency of 3750MHz, both HSDIV enabled and dividing by 6, all MSDIV dividing by 2, all LSDIV dividing by 2, and all outputs enabled as full-swing CML outputs driving 156.25MHz signals.

Note 3: VDDOx=3.3V, 1x drive strength, $f_o=250\text{MHz}$, 2pF load

Note 4: Applies to all three top-level configurations (see section 5.3). Disabling the Block B DPLL for configuration 1 saves 35-45mA on the 1.8V supply vs. the stated I_{DD18} numbers.

Table 8 - Electrical Characteristics: Non-clock CMOS Pins

Characteristics	Symbol	Min.	Typ.	Max.	Units	Notes
Input high voltage, SCL and SDA	V_{IH}	0.7 x VDD33			V	
Input low voltage, SCL and SDA	V_{IL}			0.3 x VDD33	V	
Input high voltage, all other digital inputs	V_{IH}	2.0			V	
Input low voltage, all other digital inputs	V_{IL}			0.8	V	
Input leakage current, RSTN pin	I_{ILPU}	-85		10	μA	Note 1
Input leakage current, GPIO3/IC3P pin	I_{ILGP3}	-20		20	μA	Note 1
Input leakage current, all other digital inputs	I_{IL}	-10		10	μA	Note 1
Input capacitance	C_{IN}		3	10	pF	
Input capacitance, SCL/SCLK, SDA/MOSI	C_{IN}		3	11	pF	
Input hysteresis, SCL and SDA in I ² C Bus Mode		0.05* VDD33			mV	
Output leakage (when high impedance)	I_{LO}	-10		10	μA	Note 1
Output high voltage	V_{OH}	2.4			V	$I_o = -3.0\text{mA}$
Output low voltage	V_{OL}			0.4	V	$I_o = 3.0\text{mA}$

Note 1: $0\text{V} < V_{IN} < \text{VDD33}$ for all other digital inputs.

Note 2: V_{OH} does not apply for SCL and SDA in I²C interface mode since they are open drain.

Table 9 - Electrical Characteristics: XA Clock Input

This table covers the case when there is no external crystal connected and an external oscillator or clock signal is connected to the XA pin.

Characteristics	Symbol	Min.	Typ.	Max.	Units	Notes
Input high voltage, XA	V_{IH}	1.2			V	
Input low voltage, XA	V_{IL}			0.8	V	
Input frequency on XA pin	f_{IN}	93		130	MHz	
Input leakage current	I_{IL}	-10		10	μA	
Input duty cycle		40		60	%	

Table 10 - Electrical Characteristics: Clock Inputs, ICxP/N

Characteristics	Symbol	Min.	Typ.	Max.	Units	Notes
Input voltage tolerance (each pin, single-ended)	V_{TOL}	0		VDD33	V	Note 1
Input differential voltage	$ V_{ID} $	0.1		1.4	V	Note 2
Input DC bias voltage (internally biased)	V_{CMI}		1.3		V	
Input frequency for signals going to the input block	f_{IN}	0.008		1250	MHz	Differential
		0.008		300	MHz	Single-ended
Minimum input clock high, low time, $f_{IN} \leq 250\text{MHz}$	t_H, t_L		smaller of 3ns or $0.3 \times 1/f_{IN}$		ns	Note 5
Minimum input clock high, low time, $f_{IN} > 250\text{MHz}$	t_H, t_L	0.4			ns	Note 6
Input resistance, single-ended to VDD18, ICxP or ICxN	$R_{INVDD18}$		50		$\text{k}\Omega$	
Input resistance, single-ended to VSS, ICxP or ICxN	R_{INVSS}		80		$\text{k}\Omega$	

Note 1: The device can tolerate voltages as specified in V_{TOL} w.r.t. VSS on its ICxP and ICxN pins without being damaged.

For differential input signals, proper operation of the input circuitry is only guaranteed when the other specifications in this table, including $|V_{ID}|$, are met.

Note 2: For inputs IC1P/N and IC2P/N $V_{ID} = V_{ICxP} - V_{ICxN}$. For input IC3P, $V_{ID} = V_{IC3P} - V_{CMI}$. The max V_{ID} spec only applies when a differential signal is applied on ICxP/N; it does not apply when a single-ended signal is applied on ICxP.

Note 3: **Differential signals.** The differential inputs can easily be interfaced to neighboring ICs driving LVDS, LVPECL, CML, HCSL, HSTL or other differential signal formats using a few external passive components. In general, Microsemi recommends terminating the signal with the termination/load recommended in the neighboring component's data sheet and then AC-coupling the signal into the ICxP/ICxN pins. See [Figure 20](#) for details. To connect a differential signal to IC3, AC-couple one side of the signal to IC3P and AC-couple the other side to VSS. For DC-coupling, treat the input as 1.8V CML.

Note 4: **Single-ended signals** can be connected to ICxP pins. Signals with amplitude greater than 2.5V must be DC-coupled. For signals with amplitudes less than 2.5V Microsemi recommends AC-coupling but DC-coupling can also be used. When a single-ended signal is connected to ICxP, ICxN should be connected to a capacitor (0.1 μF or 0.01 μF) to VSS.

Note 5: If [MCR1.MCSEL1=1](#) then IC1 is the DPLL's master clock source and therefore the duty cycle spec in [Table 9](#) applies to IC1 rather than the t_H, t_L spec in this table.

Note 6: The input high-speed divider must be used to divide the frequency by 2 or more.

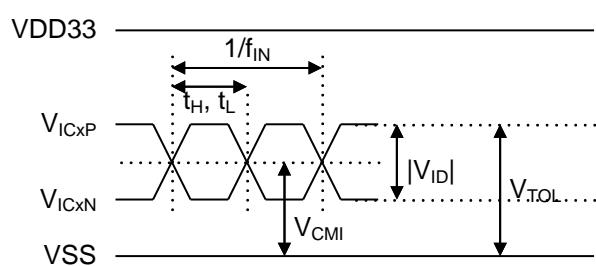


Figure 19 - Electrical Characteristics: Clock Inputs

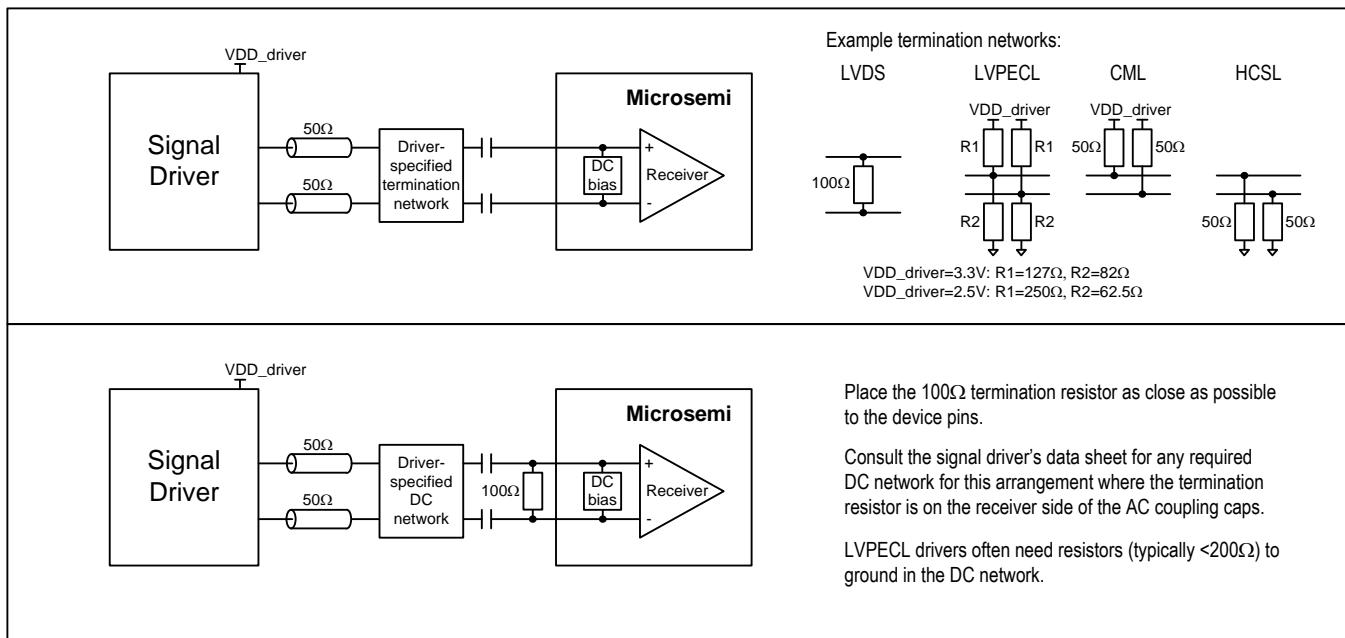


Figure 20 - Example External Components for Differential Input Signals

Table 11 - Electrical Characteristics: CML Clock Outputs

VDDOx = 3.3V±5% for CML operation.

Characteristics	Symbol	Min.	Typ.	Max.	Units	Notes
Output frequency	f_{OCML}			1035	MHz	
Output frequency from medium-speed divider	$f_{OCML,MSDIV}$			425	MHz	
Output high voltage, single-ended, OCxP or OCxN	$V_{OH,S}$		$VDDOx - 0.2$		V	
Output low voltage, single-ended, OCxP or OCxN	$V_{OL,S}$		$VDDOx - 0.6$		V	
Output common mode voltage	$V_{CM,S}$		$VDDOx - 0.4$		V	Standard Swing (OCxCR2.OCSF=1), AC coupled to 50Ω termination
Output differential voltage	$ V_{OD,S} $	320	400	500	mV	
Output differential voltage, peak-to-peak	$ V_{OD,S,PP} $	640	800	1000	mV _{P-P}	
Output high voltage, single-ended, OCxP or OCxN	$V_{OH,N}$		$VDDOx - 0.1$		V	
Output low voltage, single-ended, OCxP or OCxN	$V_{OL,N}$		$VDDOx - 0.3$		V	
Output common mode voltage	$V_{CM,N}$		$VDDOx - 0.2$		V	Narrow Swing (half the power) (OCxCR2.OCSF=2), AC coupled to 50Ω termination
Output differential voltage	$ V_{OD,N} $	160	200	250	mV	
Output differential voltage, peak-to-peak	$ V_{OD,N,PP} $	320	400	500	mV _{P-P}	
Difference in Magnitude of Differential Voltage for Complementary States	V_{DOS}			50	mV	
Output Rise/Fall Time	t_R, t_F		150		ps	20%-80%
Output Duty Cycle		45	50	55	%	Notes 2
Output Duty Cycle		40		60	%	Notes 3
Output Impedance	R_{OUT}		50		Ω	Single Ended, to VDDOx
Mismatch in a pair	ΔR_{OUT}			10	%	

Note 1: The differential CML outputs can easily be interfaced to LVDS, LVPECL, CML and other differential inputs on neighboring ICs using a few external passive components. See [Figure 22](#) for details.

Note 2: For all HSDIV, MSDIV and LSDIV combinations other than those specified in Note 3.

Note 3: For the case when [APLLCR1.HSDIV](#) specifies a half divide and [OCxCR1.MSDIV=0](#) and [OCxDIV=0](#).

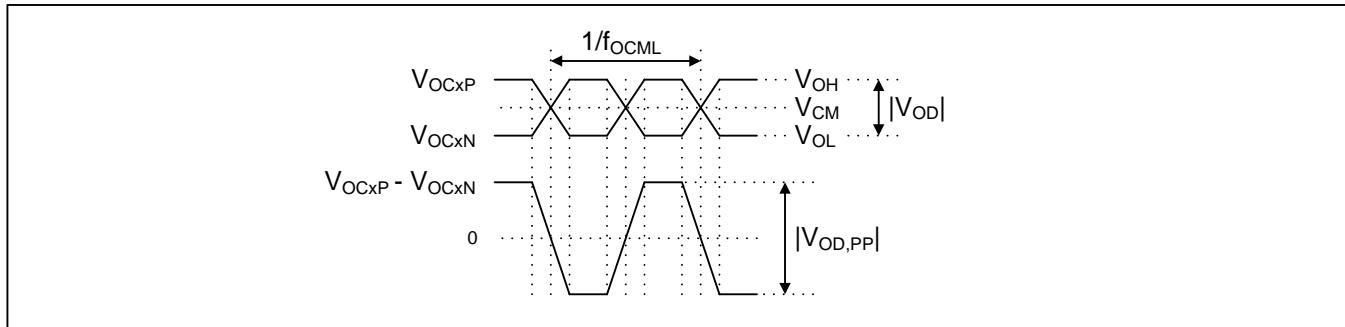


Figure 21 - Electrical Characteristics: CML Clock Outputs

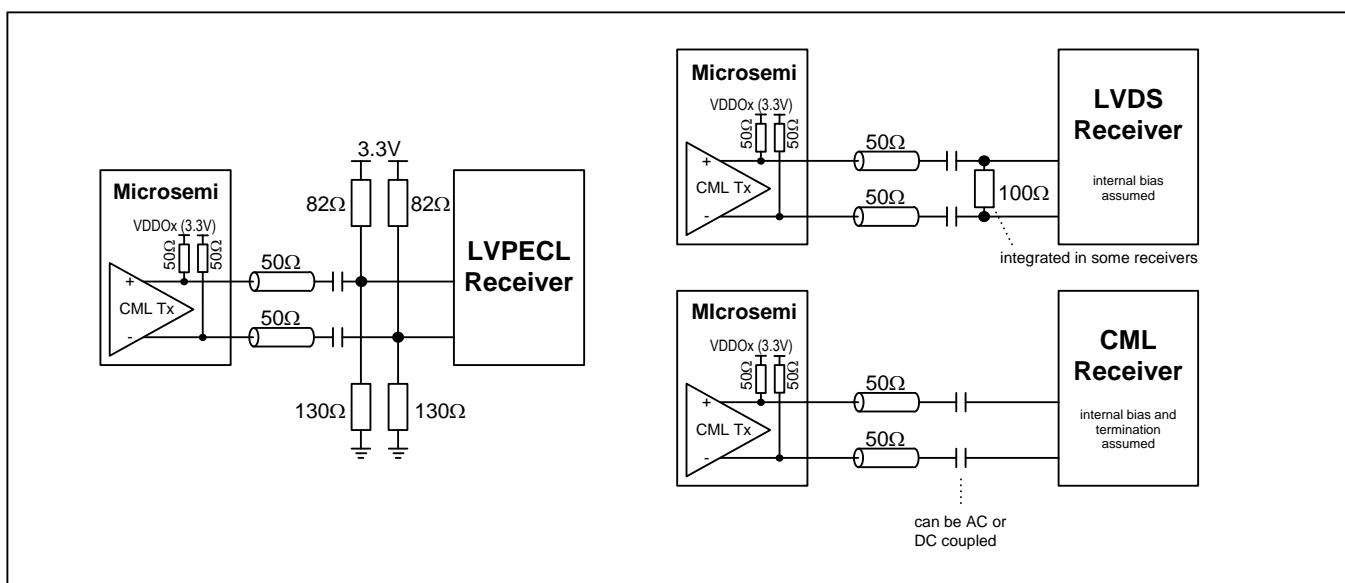


Figure 22 – Example External Components for CML Output Signals

Table 12 - Electrical Characteristics: CMOS and HSTL (Class I) Clock Outputs

Characteristics	Symbol	Min.	Typ.	Max.	Units	Notes
Output frequency	f_{OCMOS}	<<1Hz		250	MHz	Note 1
Output high voltage	V_{OH}	$VDDOx - 0.4$		$VDDOx$	V	Notes 2, 3
Output low voltage	V_{OL}	0		0.4	V	Notes 2, 3
Output rise/fall time, $VCCOx=1.8V$, $OCxCR2.DRIVE=4x$	t_R, t_F		0.4		ns	2pF load
Output rise/fall time, $VCCOx=1.8V$, $OCxCR2.DRIVE=4x$			1.2		ns	15pF load
Output rise/fall time, $VCCOx=3.3V$, $OCxCR2.DRIVE=1x$			0.7		ns	2pF load
Output rise/fall time, $VCCOx=3.3V$, $OCxCR2.DRIVE=1x$			2.2		ns	15pF load
Output duty cycle		45	50	55	%	Note 4
Output duty cycle		42	50	58	%	Notes 5, 6
Output duty cycle, $OCxNEG$ single-ended			50		%	Note 5
Output duty cycle, $OCxPOS$ single-ended			46		%	Note 5
Output current when output disabled			10		μA	$OCxCR2.OCSF=0$

Note 1: Minimum output frequency is a function of VCO frequency and output divider values and is guaranteed by design.

Note 2: For HSTL Class I, V_{OH} and V_{OL} apply for both unterminated loads and for symmetrically terminated loads, i.e. 50Ω to $VDDOx/2$.

Note 3: For $VDDOx=3.3V$ and $OCxCR2.DRIVE=1x$, $I_o=4mA$. For $VDDOx=1.5V$ and $OCxCR2.DRIVE=4x$, $I_o=8mA$.

Note 4: Output clock frequency $\leq 160MHz$ or $VDDOx \geq 1.8V$.

Note 5: Output clock frequency $> 160MHz$ and $VDDOx < 1.8V$.

Note 6: Measured differentially.

Interfacing to HCSL Components

Outputs in HSTL mode with $VDDOx=1.5V$ or $VDDOx=1.8V$ can provide an HCSL signal (V_{OH} typ. 0.75V) to a neighboring component when configured as shown in [Figure 23](#) below. For $VDDOx=1.5V$ the value of R_S should be set to 30Ω and $OCxCR2.DRIVE$ should be set to 4x. For $VDDOx=1.8V$ the value of R_S should be set to 20Ω and $OCxCR2.DRIVE$ should be set to 2x.

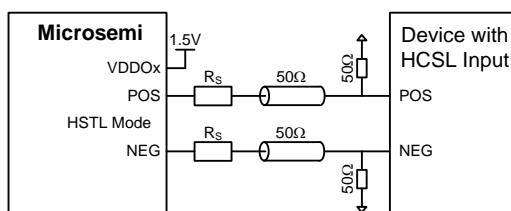


Figure 23 – Example External Components for HCSL Output Signals

Table 13 - Electrical Characteristics: APLL Frequencies

Characteristics	Symbol	Min.	Typ.	Max.	Units	Notes
APLL VCO frequency range	f_{VCO}	3715		4180	MHz	
APLL PFD input frequency	t_{PFD}	9.72		156.25	MHz	Note 1

Note 1: The TCXO frequency must be in this range for top-level configuration 1 because Block B is operated in APLL-only mode.

Table 14 - Electrical Characteristics: Jitter Specifications

Characteristics	Min.	Typ.	Max.	Units	Notes
Output Phase Jitter, 622.08MHz, Configuration 1		Depends on TCXO jitter		ps RMS	Notes 1, 2
Output Phase Jitter, 622.08MHz, Configuration 2		0.34	0.45	ps RMS	Notes 1, 2
Output Phase Jitter, 622.08MHz, Configuration 3		0.24	0.3	ps RMS	Notes 1, 2
Output Period Jitter		9.5		ps pk-pk	N=10000, Note 4
Output Half-Period Jitter		25		ps pk-pk	N=10000, Note 4
Output Cycle-to-Cycle Jitter		8.5		ps pk	N=10000, Note 4
Jitter Transfer Bandwidth, Block A DPLL		Programmable: 0.1 to 10Hz		Hz	
Jitter Transfer Bandwidth, Block B, Jitter Attenuation		Programmable: 20 to 100Hz		Hz	
Jitter Transfer Bandwidth, Block B, Frequency Multiplication		600		kHz	Note 3

Note 1: Jitter calculated from integrated phase noise from 12kHz to 20MHz.

Note 2: Tested with 114.285MHz XO (Vectron VCC1-1537-114M285) connected to XA_B, APLL VCO frequency 3732.48MHz, HSDIV1=6, OC1 frequency 622.08MHz.

Note 3: APLL bandwidth and damping factor can be field configured over a limited range. Contact the factory for details.

Note 4: Outputs from a half-divide (e.g. 4.5) in the high-speed divider followed by only an odd divide in the medium-speed divider can have higher jitter values. Example: 100MHz from VCO=3750MHz, HSDIV2=7.5, MSDIV=5 has typical period jitter of 20ps (rising edge) or 55ps (falling edge), typical cycle-to-cycle jitter of 19ps and typical half-period jitter of 77ps.

Table 15 - Electrical Characteristics: Typical Output Jitter Performance, Configuration 2

Output Frequency	Output Jitter, ps RMS 114.285MHz XO ¹
625MHz	0.34
156.25MHz	0.35
125MHz	0.37
622.08MHz	0.34
155.52MHz	0.35
622.08MHz * 255/237	0.36
155.52MHz * 255/237	0.36
614.4MHz	0.35
153.6MHz	0.36

Note 1: DPLL Locked to 25MHz input on IC1, APLL locked to DPLL, Vectron VCC1-1537-114M285 on XA_B.

Note 2: All signals are differential unless otherwise stated. Jitter is integrated 12kHz to 5MHz for 25MHz output frequency and 12kHz to 20MHz for all other output frequencies.

Table 16 - Electrical Characteristics: Typical Output Jitter Performance, Configuration 3

Output Frequency	Output Jitter, ps RMS 114.285MHz XO ¹
625MHz	0.24
156.25MHz	0.265
125MHz	0.30
622.08MHz	0.24
155.52MHz	0.27
622.08MHz * 255/237	0.26
155.52MHz * 255/237	0.28
614.4MHz	0.25
153.6MHz	0.27

Note 1: DPLL Locked to 25MHz input on IC1, APLL locked to DPLL, Vectron VCC1-1537-114M285 on XA_B.

Note 2: All signals are differential unless otherwise stated. Jitter is integrated 12kHz to 5MHz for 25MHz output frequency and 12kHz to 20MHz for all other output frequencies.

Table 17 - Electrical Characteristics: Typical Input-to-Output Clock Delay

Mode	Delay, Input Clock Edge to Output Clock Edge
Any	<p>Using external feedback: 0 (DPLL's DPHOFF field can be used to compensate for board propagation delays. See section 5.10.4.3.1.)</p> <p>Without using external feedback: Non deterministic but constant as long as the DPLL and APLL remain locked and output clock phases are not adjusted as described in section 5.10.4.1. See section 5.10.4.3.2.</p> <p>External feedback configuration has a DPLL-sourced output clock wired externally to an ICx input clock and ICSCR1.FBSEL≠0. In this configuration the DPLL drives the output clock phase to be equal to the DPLL's input clock phase + DPHOFF.</p> <p>Delay can be tuned for all outputs traceable to the DPLL by adjusting the DPLL's DPHOFF field. Delay for an individual output can be tuned using the output phase adjustment capability described in section 5.10.4.1.</p>

Table 18 - Electrical Characteristics: Typical Output-to-Output Clock Delay

Mode	Delay, Output Clock Edge to Output Clock Edge
Any	<p><100ps</p> <p>Requires phase adjustment and phase alignment capability described in section 5.10.4.</p>

Table 19 - Electrical Characteristics: SPI Slave Interface Timing, Device Registers

Characteristics (Note 1, 2)	Symbol	Min.	Typ.	Max.	Units	Notes
SCLK frequency	f_{BUS}			10	MHz	
SCLK cycle time	t_{CYC}	100			ns	
CSN setup to first SCLK edge	t_{SUC}	50			ns	
CSN hold time after last SCLK edge	t_{HDC}	50			ns	
CSN high time	t_{CSH}	50			ns	
SCLK high time	t_{CLKH}	40			ns	
SCLK low time	t_{CLKL}	40			ns	
MOSI data setup time	t_{SUI}	10			ns	
MOSI data hold time	t_{HDI}	10			ns	
MISO enable time from SCLK edge	t_{EN}	0			ns	
MISO disable time from CSN high	t_{DIS}			80	ns	
MISO data valid time	t_{DV}			40	ns	
MISO data hold time from SCLK edge	t_{HDO}	0			ns	
CSN, MOSI input rise time, fall time	t_R, t_F			10	ns	

Note 1: All timing is specified with 100pF load on all SPI pins.

Note 2: All parameters in this table are guaranteed by design or characterization.

Note 3: See timing diagram in [Figure 24](#).

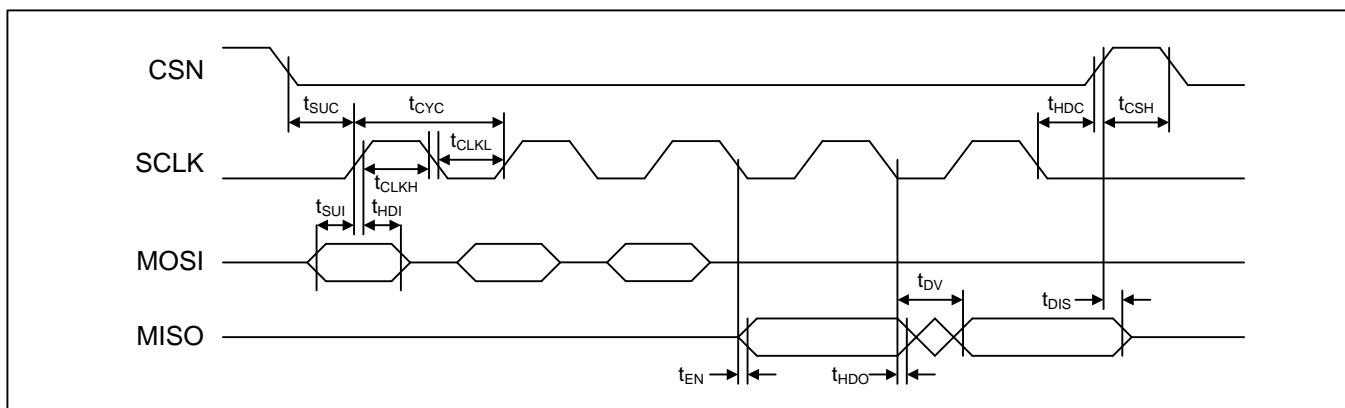


Figure 24 - SPI Interface Timing

Table 20 - Electrical Characteristics: SPI Slave Interface Timing, Internal EEPROM

Characteristics (Note 1, 2)	Symbol	Min.	Typ.	Max.	Units	Notes
SCLK frequency	f_{BUS}			10	MHz	
SCLK cycle time	t_{CYC}	100			ns	
CSN setup to first SCLK edge	t_{SUC}	50			ns	
CSN hold time after last SCLK edge	t_{HDC}	51			ns	
CSN high time	t_{CSH}	51			ns	
SCLK high time	t_{CLKH}	41			ns	
SCLK low time	t_{CLKL}	41			ns	
MOSI data setup time	t_{SUI}	11			ns	
MOSI data hold time	t_{HDI}	11			ns	
MISO enable time from SCLK edge	t_{EN}	0			ns	
MISO disable time from CSN high	t_{DIS}			90	ns	
MISO data valid time	t_{DV}			60	ns	
MISO data hold time from SCLK edge	t_{HDO}	0			ns	
CSN, MOSI input rise time, fall time	t_R, t_F			10	ns	

Note 1: This timing applies (a) when [EESEL](#)=1 and (b) in direct EEPROM write mode (see section [5.15.2](#)).

Note 2: All timing is specified with 100pF load on all SPI pins.

Note 3: All parameters in this table are guaranteed by design or characterization.

Note 4: See timing diagram in [Figure 24](#).

Table 21 - Electrical Characteristics: I²C Slave Interface Timing

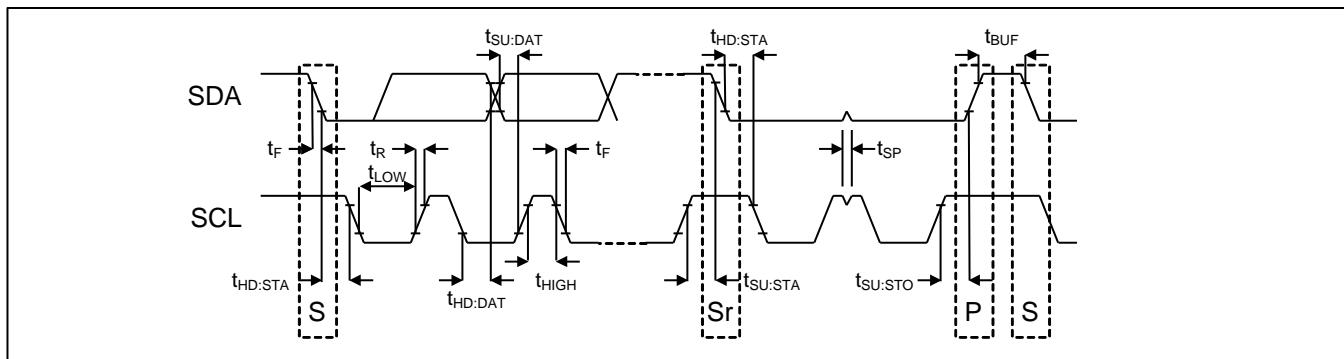
Characteristics	Symbol	Min.	Typ.	Max.	Units	Notes
SCL clock frequency	f_{SCL}			400	kHz	Note 1
Hold time, START condition	$t_{HD:STA}$	0.6			μs	
Low time, SCL	t_{LOW}	1.3			μs	
High time, SCL	t_{HIGH}	0.6			μs	
Setup time, START condition	$t_{SU:STA}$	0.6			μs	
Data hold time	$t_{HD:DAT}$	0		0.9	μs	Notes 2 and 3
Data setup time	$t_{SU:DAT}$	100			ns	
Rise time	t_R				ns	Note 4
Fall time	t_F	$20 + 0.1C_b$		300	ns	C_b is cap. of one bus line
Setup time, STOP condition	$t_{SU:STO}$	0.6			μs	
Bus free time between STOP/START	t_{BUF}	1.3			μs	
Pulse width of spikes which must be suppressed by the input filter	t_{SP}	0		50	ns	

Note 1: The timing parameters in this table are specifically for 400kbps Fast Mode. Fast Mode devices are downward-compatible with 100kbps Standard Mode I²C bus timing. All parameters in this table are guaranteed by design or characterization. All values referred to V_{IHmin} and V_{ILmax} levels (see Table 8).

Note 2: The device internally provides a hold time of at least 300ns for the SDA signal (referred to the V_{IHmin} of the SCL signal) to bridge the undefined region of the falling edge of SCL. Other devices must provide this hold time as well per the I²C specification.

Note 3: The I²C specification indicates that the maximum $t_{HD:DAT}$ spec only has to be met if the device does not stretch the low period (t_{LOW}) of the SCL signal. The device does not stretch the low period of the SCL signal.

Note 4: Determined by choice of pull-up resistor.

Figure 25 - I²C Slave Interface Timing

8. Package and Thermal Information

8.1 Package Top Mark Format

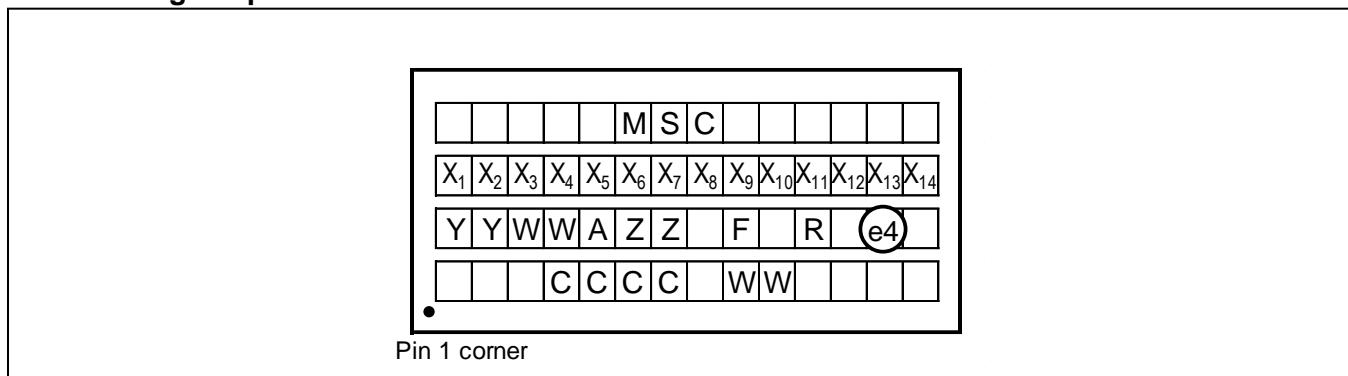


Figure 26 - Non-customized Device Top Mark

Table 22 – Package Top Mark Legend

Line	Characters	Description
2	X ₁ – X ₁₄	Part Number
3	YY	Last Two Digits of the Year of Encapsulation
3	WW	Work Week of Assembly
3	A	Assembly Location Code
3	ZZ	Assembly Lot Sequence
3	F	Fab Code
3	R	Product Revision Code
3	e4	Denotes RoHS Level of Package
4	CCID	Custom Programming Identification Code only present for devices with EEPROM memory custom-programmed by Microsemi
4	WW	Work Week of Custom Programming only present for devices with EEPROM memory custom-programmed by Microsemi

8.2 Thermal Specifications

Table 23 - 5x10mm LGA Package Thermal Properties

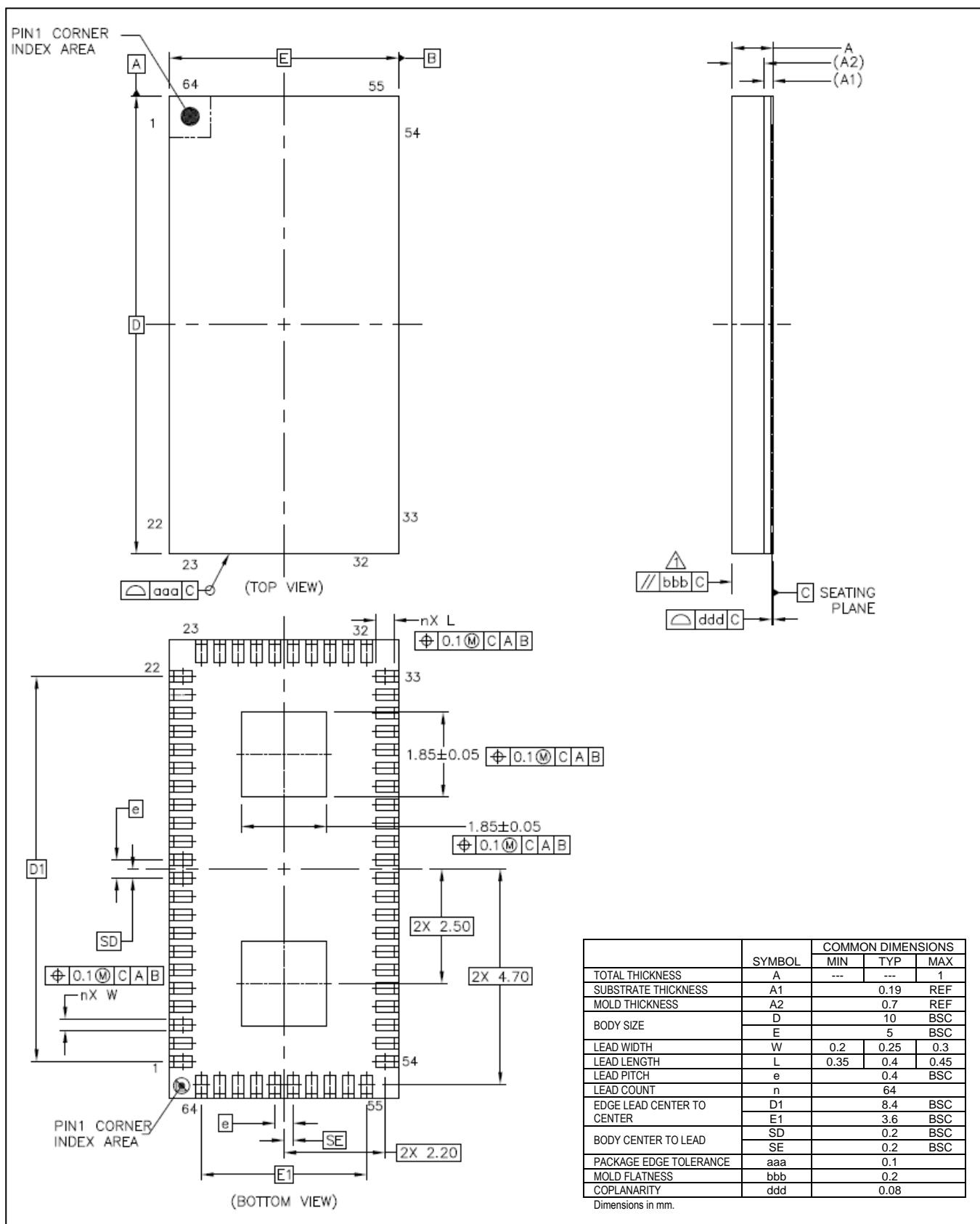
PARAMETER	SYMBOL	CONDITIONS	VALUE	UNITS
Maximum Ambient Temperature	T_A		85	°C
Maximum Junction Temperature	T_{JMAX}		125	°C
Junction to Ambient Thermal Resistance (Note 1)	θ_{JA}	still air	15.68	°C/W
		1m/s airflow	13.22	
		2.5m/s airflow	12.32	
Junction to Board Thermal Resistance	θ_{JB}		6.87	°C/W
Junction to Case Thermal Resistance	θ_{JC}		4.28	°C/W
Junction to Pad Thermal Resistance (Note 2)	θ_{JP}	still air	3.99	°C/W
Junction to Top-Center Thermal Characterization Parameter	ψ_{JT}	still air	1.18	°C/W
		1m/s airflow	1.29	
		2.5m/s airflow	1.38	

Note 1: Theta-JA (θ_{JA}) is the thermal resistance from junction to ambient when the package is mounted on an 8-layer standard test board and dissipating maximum power.

Note 2: Theta-JP (θ_{JP}) is the thermal resistance from junction to the exposed pads on the bottom of the package.

Note 3: For all numbers in the table, each exposed pad is connected to the ground plane with a 3x3 array of thermal vias; via diameter 0.2mm.

9. Mechanical Drawing



10. Acronyms and Abbreviations

APLL	analog phase locked loop
BITS	building integrated timing supply
CML	current mode logic
DFS	digital frequency synthesis
DPLL	digital phase locked loop
EEC	Ethernet equipment clock
GbE	gigabit Ethernet
HCSL	high-speed current steering logic
HSTL	high-speed transceiver logic
I/O	input/output
LVDS	low-voltage differential signal
LVPECL	low-voltage positive emitter-coupled logic
MTIE	maximum time interval error
OCXO	oven controlled crystal oscillator
PFD	phase/frequency detector
PLL	phase locked loop
ppb	parts per billion
ppm	parts per million
pk-pk	peak-to-peak
RMS	root-mean-square
RO	read-only
R/W	read/write
SDH	synchronous digital hierarchy
SEC	SDH equipment clock
SETS	synchronous equipment timing source
SONET	synchronous optical network
SSU	synchronization supply unit
STM	synchronous transport module
TDEV	time deviation
TCXO	temperature-compensated crystal oscillator
UI	unit interval
UI _{PP} or UI _{P-P}	unit interval, peak to peak
XO	crystal oscillator

11. Data Sheet Revision History

Revision	Description
09-Oct-2015	First general release
17-Dec-2015	In Table 14 added max jitter numbers for configurations 2 and 3. In Table 15 and Table 16 deleted Note 2 because it did not apply.
21-Apr-2016	Added Note 3 to Table 23 . In Table 10 changed t_H , t_L specs for $f_{IN} > 250\text{MHz}$ from 0.4/ f_{IN} typical to 0.4ns min and added Note 6. Edited section 5.13 to reduce RSTN assertion time for all but one specific situation. Added a note to the MCR1.RST description to indicate the need to clear the MCSEL, MCSEL1 and ROSCD bits before setting the RST bit. In section 5.11.2 in the Read Transactions paragraph added a note describing the case where the I ² C write is separated from the I ² C read by other transactions. In section 5.5.1 added a sentence about connecting one oscillator to both channels' XA pins. Corrected section 5.3.3 to indicate higher bandwidth for bigger thermal transients.
25-Jul-2016	Documented DPLLCR2.HSEN . Documented the PHLKTO and LKATO registers.
23-Jan-2018	In section 5.10.2 second paragraph added maximum input frequency sentences for the medium-speed and low-speed dividers. In section 6 changed the end of the address range from 0x1FF to 0x6FF to include test registers. In the DPHASE and DPHOFF register descriptions corrected units from ps to 2^{-10}ns and indicated that positive values represent phase earlier in time. In the EESEL register description, added a note that the EESEL bit is write-only. In Table 1 , in the SCL/SCLK and SDA/MOSI pin descriptions added notes indicating the need for an external pullup resistor for I ² C operation and referring the reader to the I ² C specification for details. In OCxPH register description, indicated that positive values represent later in time. In the APLLCSR4 register description indicated that "increment" moves the signal later in time and "decrement" moves the signal earlier in time.



Microsemi Corporate Headquarters
One Enterprise
Aliso Viejo, CA 92656 USA

Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996

E-mail: sales.support@microsemi.com

©2018 Microsemi Corporation. All rights reserved. Microsemi and the Microsemi logo are trademarks of Microsemi Corporation. All other trademarks and service marks are the property of their respective owners.

Microsemi Corporation (Nasdaq: MSCC) offers a comprehensive portfolio of semiconductor and system solutions for communications, defense & security, aerospace and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; security technologies and scalable anti-tamper products; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif., and has approximately 3,400 employees globally. Learn more at www.microsemi.com.

Microsemi makes no warranty, representation, or guarantee regarding the information contained herein or the suitability of its products and services for any particular purpose, nor does Microsemi assume any liability whatsoever arising out of the application or use of any product or circuit. The products sold hereunder and any other products sold by Microsemi have been subject to limited testing and should not be used in conjunction with mission-critical equipment or applications. Any performance specifications are believed to be reliable but are not verified, and Buyer must conduct and complete all performance and other testing of the products, alone and together with, or installed in, any end-products. Buyer shall not rely on any data and performance specifications or parameters provided by Microsemi. It is the Buyer's responsibility to independently determine suitability of any products and to test and verify the same. The information provided by Microsemi hereunder is provided "as is, where is" and with all faults, and the entire risk associated with such information is entirely with the Buyer. Microsemi does not grant, explicitly or implicitly, to any party any patent rights, licenses, or any other IP rights, whether with regard to such information itself or anything described by such information. Information provided in this document is proprietary to Microsemi, and Microsemi reserves the right to make any changes to the information in this document or to any products and services at any time without notice.