BSC9132 QDS Board Reference Manual

Devices Supported
BSC9132

BSC9132QDSRM
Rev 0
05/2014

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support

Document Number: BSC9132QDSRM
Agile Number: 926-78847 Rev B

Contents

Contents

Chapter 3
Architecture-Qixis

Contents

Contents

Contents

Chapter 6
Revision History

Contents

Tables

Tables

Tables

Tables

Figures

Figures

Figures

Figures

Chapter 1
Overview

The BSC9132 QorIQ Development System (QDS) is a high-performance computing evaluation, development, and test platform supporting the **BSC9132** QorIQ Power Architecture® processor.

The BSC9132 QDS Board Reference Manual is optimized to support the high-bandwidth memory port, as well as the highly-configurable SerDes ports. The BSC9132 QDS Board Reference Manual is designed for a standard ATX form-factor, allowing it to be shipped in an off-the-shelf ATX chassis, if needed. The system is lead-free and RoHS-compliant.

1.1 Related documentation

For information on products and resources used with the BSC9132 QDS, use the references listed in Table 1-1.

Some of these documents may be available only under a non-disclosure agreement (NDA). To request access to these documents, contact your local field applications engineer or sales representative.

Table 1-1. Related Documentation

Document	Description
BSC9132 QorIQ Qonverge Multicore Baseband Processor Data Sheet (document BSC9132EC)	Provides information about Pin assignments, Electrical, characteristics, Hardware design, considerations, Package information, and Ordering information.
BSC9132 QorIQ Integrated Multicore Communication Processor Family Reference Manual (BSC9132RM)	Provides a detailed description about T2080 QorIQ multicore processor, and features, such as memory map, serial interfaces, power supply, chip features, and clock information.
The SystemID Format for Power Architecture™ Development Systems (AN3638)	Freescale Semiconductor Power Architecture™ technology-based evaluation and development platforms may optionally implement a “System ID” non-volatile memory device. This device stores important configuration data about the board.

2 Acronyms and abbreviation

The table below lists and explains the acronyms and abbreviations used in this document.

Table 2-2. Acronyms and abbreviations

Usage	Description
ADDR	Address
ATX	Advanced Technology Extended (power supply)
Komodo	Test Board for high Speed Path
AUX	Auxiliary
BRDCFG	Board Configuration
BVDD	IFC Bus Direct Current Voltage
CFG	Configuration
CLK	Clock
CLKIN	Clock Input (interchangeable with SYSCLK)
COP	Common On-Chip Processor
CVDD	Clock Driver Supply Voltage / Bus Control Voltage
DDR	Double Data Rate
DIP	Dual-In-Line Package (switches)
DRAM	Dynamic Random Access Memory
QDS	Qualification Development System
EC	Chip HW Specification
ECC	Error Detection and Correction
EEPROM	Electrically Erasable Programmable ROM
eSDHC	Enhanced Secure Digital High Capacity Card
ETH	Ethernet
evt	event
FS	Frequency Select
FCM	NAND Flash Control Machine
FLASH	Flash Memory Chip
FPGA	Field Programmable Gate Array
GETH	Giga Ethernet (GbE)
GPIO	General Purpose In/Out
GVDD	DDR Supply voltage
Host	BSC9132
HRESET	Hard Reset

Table 2-2. Acronyms and abbreviations

Usage	Description
I^2C	Inter-Integrated Circuit Multi-Master Serial Computer Bus
ID	Identification
IDE	Integrated Development Environment
IO	Input/Output
IPL	Initial Program Load
ISO	Isolated
JTAG	Joint Test Access Group (IEEE® Std. 1149.1™)
LBMAP	Local Bus Map
LED/LD	Light-emitting Diode
LSB	Least Significant Bit
LVDD	BSC9132 QDS GETH (Low) Voltage
MSB	Most Significant Bit
MUX	Multiplexer
NAND	Flash Memory
QIXIS	FPGA Block Logic Design
NOR	Flash Memory
DCM	Off-line Configuration Manager (FPGA-embedded)
OVDD	Output Voltage
PCIe/PEX	PCIe = PCI Express = PEX
PG	Power Good
PHY	Physical Layer
WP	Write Protect
PLL	Phased Lock Loop
POVDD	Parameter Operating Voltage
ppm	Parts per Million
PROC ISO	Processor Isolated
PROMJet	Memory Emulator by EmuTec Inc.
PROMJet Flash	Flash by EmuTec Inc.
PS	Power Supply
PWR	Power

Table 2-2. Acronyms and abbreviations

Usage	Description
QorIQ	Brand of power architecture based on a Freescale communications micro controller.
RC	Root Complex
RCW	Reset Configuration Word
REF	Reference
REF CLK	Reference Clock (Clock Synthesizer Input Value)
REG	Register
REG CFG	Configuration Register
REQ	Request
ROM	Read Only Memory
RST	Reset
RTC	Real-time Clock
SD	Secure Digital Card
SDHC	Secure Digital High Capacity
SDREFCLK	SerDes Reference Clock
SEL	Select
SerDes (SRDS)	Serializer/Deserializer; such as PEX, SGMII,CPRI
SGMII	Serial Gigabit Media Independent Interface
SMA	Subminiature Version B Connector
SPD	Speed
SRAM	Static Random Access Memory
STAT	Status
SVDD	Supply Voltage
SVR	System Version
SW	Switch
SYSCLK	System Clock
TAP	Telocator Alphanumeric Protocol; such as USB TAP or ETH TAP
TESTSEL	Test Select
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus

Table 2-2. Acronyms and abbreviations

Usage	Description
USBCLK	USB Clock
WP	Write Protect
CPRO	Public Radio Interface
SGMII	Serial GEthernet
XVDD	Phased Lock Loop Voltage

2.1 Features

The BSC9132 processor includes the following functions and features:

- Two e500-mc Power Architecture cores, each with a backside 512-KB L2 Cache with ECC
  - Three levels of instructions: user, supervisor, and hypervisor
  - Independent boot and reset
  - Secure boot capability
- Two StarCore SC3850 DSP subsystems, each with a 512-KB private L2 caches
- 32-KB of shared M3 memory
- The Multi Accelerator Platform Engine for Pico BaseStation Baseband Processing (MAPLE-B2P)
  - A multi-standard baseband algorithm accelerator for Channel Decoding/Encoding Fourier
  - Transforms UMTS chip rate processing, LTE UP/DL Channel processing, and CRC algorithms
  - 800 MHz to 1 GHz clock frequency
- Two DDR3/3L memory interfaces with 32-bit data width (40 bits including ECC), up to 1333 MHz data rate
  - data rate 32 KB 8-way level 1 data/instruction cache (L1 Dcache/ICache)
  - 512 KB 8-way level 2 unified instruction/data cache (L2 cache/M2 memory)
  - Memory management unit (MMU)
  - Enhanced programmable interrupt controller (EPIC)
  - Debug and profiling unit (DPU)
  - Two 32-bit timers
- Dedicated security engine featuring trusted
- Two DMA controllers
  - OCNDMA with four bidirectional channels
  - Sys DMA with sixteen bidirectional channels
  - IEEE 1588™ v2 support
- Interfaces
  - Two triple-speed Gigabit Ethernet controllers featuring network acceleration including - IEEE 1588™ v2 hardware support for two SGMII ports and virtualization (VeTSEC)

  - PCI Express controller, which complies with the PCI Express™ Base Specification Revision 2.0
  - Two Common Public Radio Interface (CPRI) controller lanes
  - Antenna Interface Controller (AIC), supporting four industry standard JESD207/four custom
    - ADI RF interfaces (three dual port and one single port) and a 2-lane CPRI interface
    - ADI lanes support both full duplex FDD support and half duplex TDD support
  - Universal Subscriber Identity Module (USIM) interface that facilitates communication to SIM
  - Multicore Programmable Interrupt Controller
  - Two I^2C controllers
  - Four DUART
  - Integrated Flash memory controller (IFC), supporting NAND, NOR, ASIC and GPMC.
  - Two PCI Express 2.0 controllers/ports
  - Two enhanced Serial Peripheral Interfaces (eSPI)
  - High-speed USB controller (USB 2.0)
    - Host and device support
    - ULPI interface to PHY
  - Sixteen 32-bit timers
- The two e500 cores subsystems within Power Architecture consist of the following:
  - Programmable interrupt controller (PIC) compliant with OpenPIC standard
  - 32-KB L1 instruction cache
  - Shared 512-KB L2 cache/L2 memory/L2 stash32-KB L1 data cache
  - Timers
- Each SC3850 core subsystem consists of the following:
  - 32 KB 8-way level 1 instruction cache (L1 ICache)
  - 32 KB 8-way level 1 data cache (L1 DCache)
  - 512 KB 8-way level 2 unified instruction/data cache (M2 memory)
  - Memory Management Unit (MMU)
  - Enhanced programmable interrupt controller (EPIC)
  - Debug and profiling unit (DPU)
  - Two 32-bit timers
- **QIXIS** System Logic FPGA
  - Manages system power and reset sequencing
  - Manages system, for DSP and DDR clock speed selections
  - Manages dynamic reconfiguration
  - Internal processor (GMSA) supports background data collection on voltage, power, and temperature
  - Registers allow full control of all device features
  - Support for classic test features, including:

  - POSt
  - IRS
  - Synchronous signal assertion (resets, IRQs)
- System fault monitoring and status display through LED's system fault monitoring
- Runs from ATX “hot” power rails allowing FPGA operation while system is off.

- SERDES Connections
  - Supports one 2xPCIe express channel
    - On-board 16xPCIe slot
  - Supports two CPRI channels configurations
    - On-board two SFP + optical link transceivers
  - Supports two SGMII interface links
    - On-board tow SGMII Vitesse VSC8221PHY devices
- Clocks
  - System clock (SYSCLK), DSP clock (DSPCLK) and DDR clock (DDRCLK)
    - Switch selectable to one of 4 common settings in the interval 66MHz-133MHz
    - Software selectable in 1MHz increments from 1-200MHz
  - SERDES clocks
    - Supports 100, 125 MHz for SD1 and 100, 125, 122.88 MHz clocking for SD2 port.
  - 125MHz Ethernet clock
  - 2.048Mhz TDM Clock
  - 16 MHz RTC Clock
- Power Supplies
  - Dedicated regulators for BVDD and XVDD, XPAD voltage rails
  - Dedicated regulators for G1VDD(VTT1/VREF1) and G2VDD(VTT2/VREF2) (1.5V/1.35V and 0.75V/0.675V)
  - Other regulators for CVDD, XVDD, OVDD, LVDD, general 1.5V, 1.8V and 2.5V power for peripherals

2.2 Block diagram

This section provides a high-level overview of the **BSC9132**. Figure 2-1 shows the major functional units within the device and Figure 2-2 shows the overall architecture of the BSC9132 QDS platform.

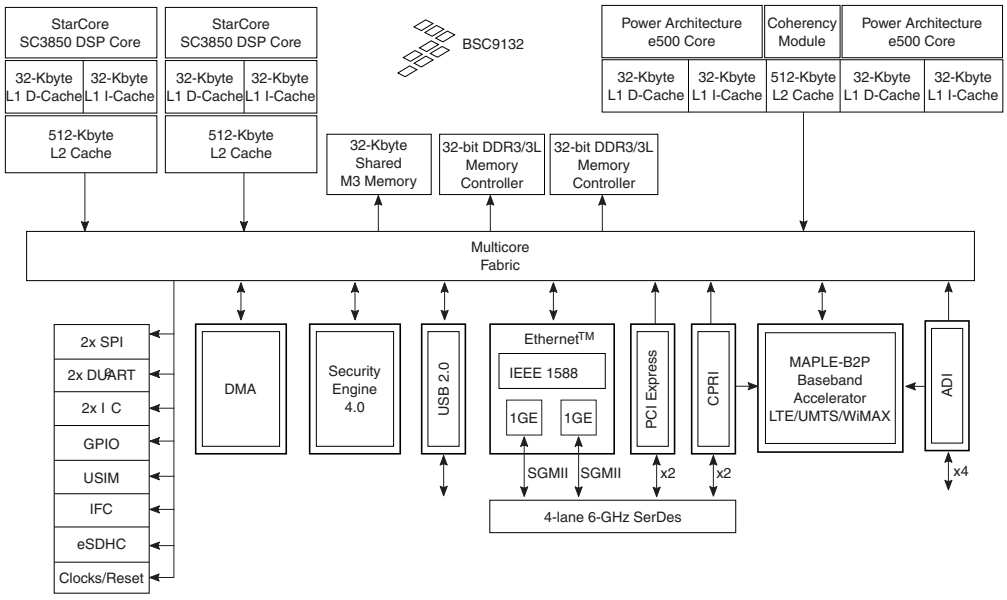


Figure 2-1. BSC9132 QorIQ block diagram

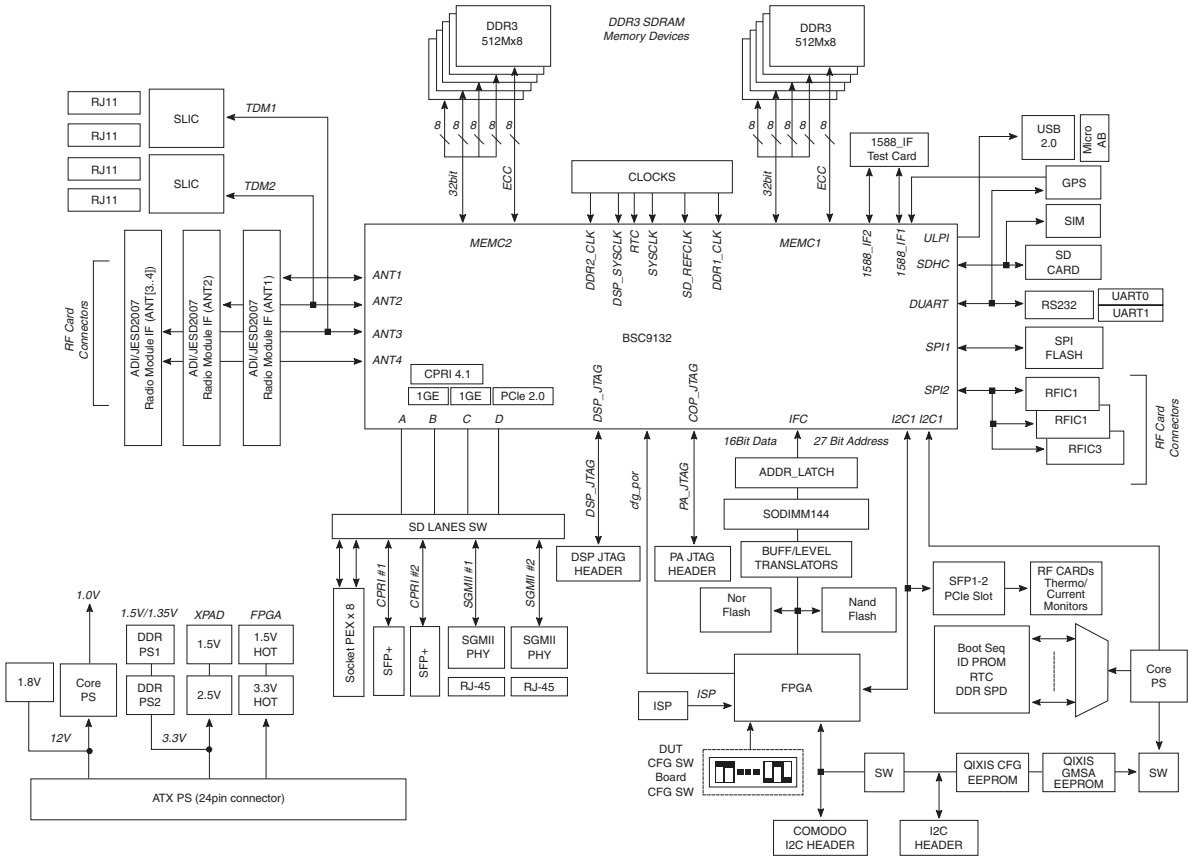


Figure 2-2. BSC9132 QDS block diagram

NOTE

To examine the details of the BSC9132 QDS board top view, you need to zoom-in the image.

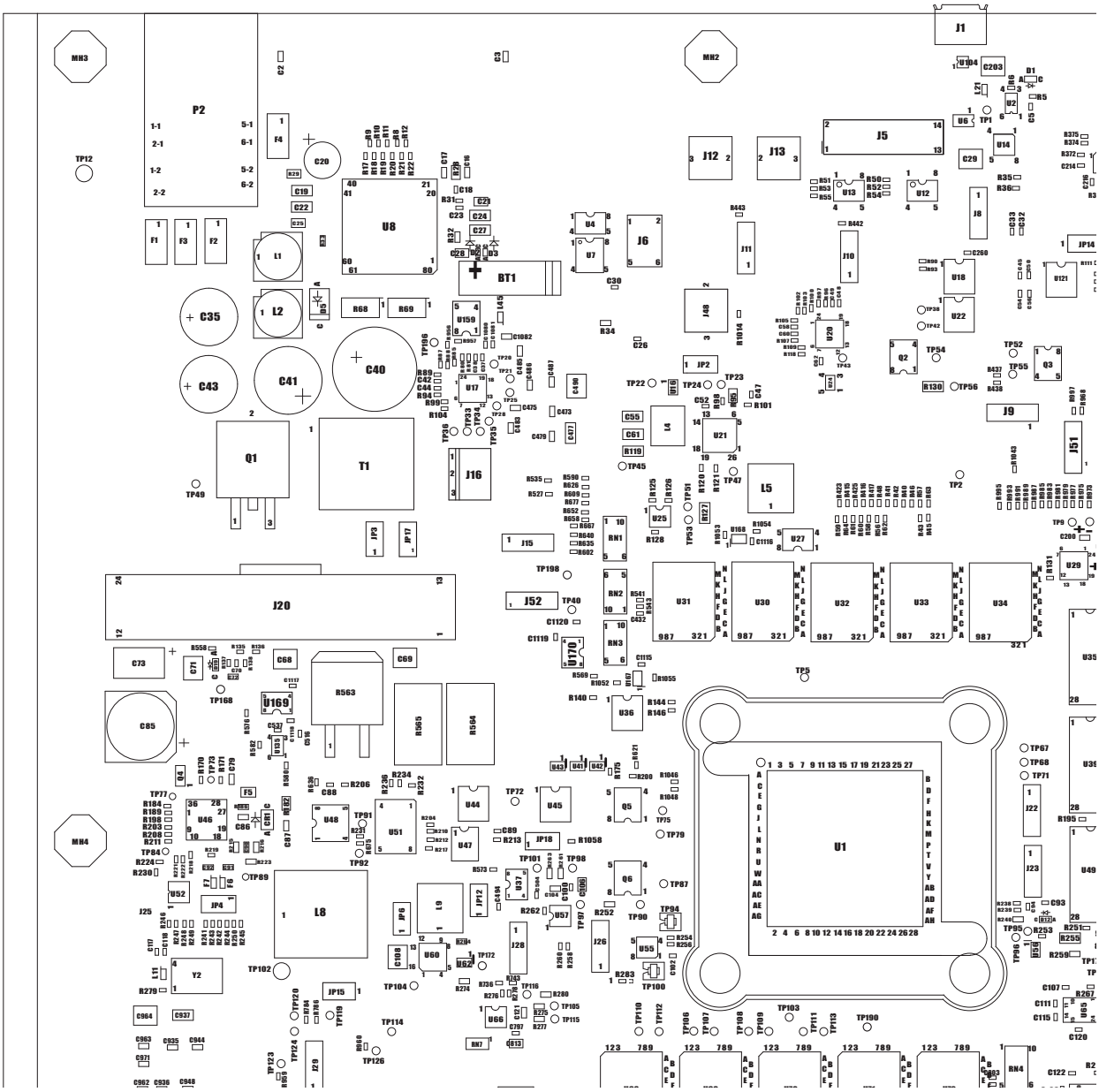


Figure 2-3. BSC9132 QDS board top view

2.3 Test environment

For general hardware and/or software development and evaluation, the **BSC9132 QDS** provides a number of ways to communicate with the DUT based on the flow.

In particular, features dedicated to in-depth silicon evaluation are disabled by default. As an example, for test purposes, systems halt during reset or restart and wait for permission from a remote test controller before proceeding - such behavior is not desirable during software development, or even when operating as a server.

2.3.1 Validation flow

For validation team to use the board through Petra flow use the following steps:

1. Insert the processor/interposer card at the socket.
2. Insert the Komodo card in PCIe slot J39.

NOTE

It is possible to make a communication link either by using cable or from komodo card to the J19 connector. Ensure that pin 1 gets connected correctly as per the silk screen marking.

3. Change the processor POR settings through DIP switches, if needed.
4. Connect the ATX Power cable at connector J1 and power on ATX power supply.
5. The FPGA would power up the processor and the board. Wait for the following LED's to light up:
   a) DUT_PG
   b) ACTIVE
   c) READY

NOTE

If any other LED is lit, then check the board debug section to understand the issues.

6. Using the test port slot or the COP JTAG header J24, access the processor.
7. For Dual DUT setup, use Card (A) setup as mentioned above in steps 1 to 6.
8. For Card B, perform the following setup (you do not need another Komodo card):
   a) Another test Port card needs to be inserted on the Card B with the PEX cable connecting the Card A komodo Card.
   b) Connector J500 header to connect the I2C Cable of Card A with J5 of Card B.
   c) Using the dip switches change the FPGA (U16) and I2C multiplexer I2C addresses as needed so that no address conflict happens.
   d) A separate power supply is also needed for powering Card B.

2.3.2 COP/JTAG header flow

To use the board, through the COP header use the following steps:

1. Insert the processor/interposer card at the socket.
2. Change the processor POR settings through DIP switches if needed.
3. Use J24 to connect debugger to the COP header.
4. Connect the ATX Power cable at connector J1 and power on ATX power supply.
5. The FPGA would power up the processor and the board. Wait for the following LED's to light up:
   a) DUT_PG
   b) ACTIVE

c) READY

If any other LED is lit, then check the board debug section to understand the issues.

2.3.3 Evaluation test system use

For test use, the system is configured into the test mode and QIXIS controls the target on behalf of a remote controller, typically a Komodo or another I2C-based system. The figure below shows a general overview of QIXIS residing in a typical system.

Figure 2-4 shows a general overview of **QIXIS** residing in a typical system:

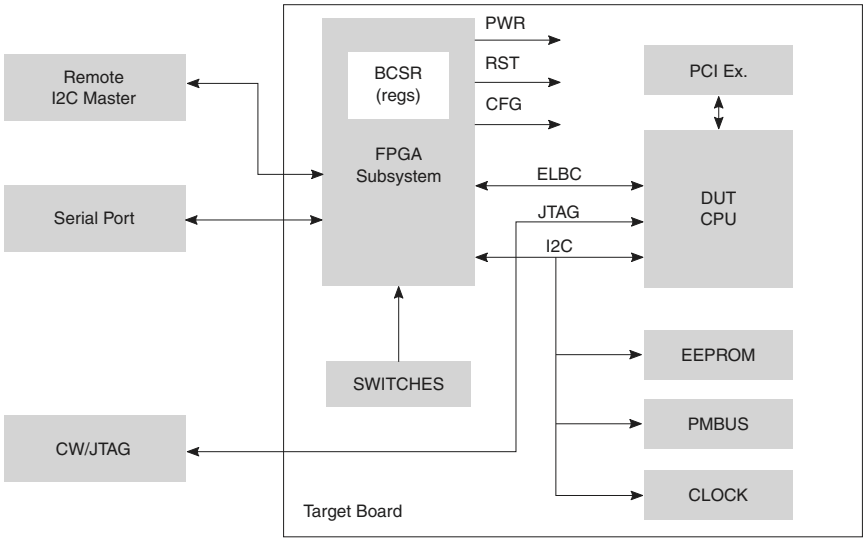


Figure 2-4. BSC9132 QDS QIXIS FPGA control paths

With such an arrangement, you can:

- Restart the target system, halting before releasing the DUT (Device Under Test)
- Download (via I2C, or a remote I2C system) any or all of the following:
  - DUT configuration values
  - Board configuration values
  - RCW (Reset Control Word) values
  - RCW location (internal to **QIXIS** or external)
  - Target memory values
  - Clock speeds
  - and numerous others
- Allow the DUT to begin execution in the new test environment
- Monitor DUT operation, halting it, if necessary

- Collect test results via:
  - QIXIS dedicated registers
  - DUT memory

Table 2-3. Test evaluation methods

Test mode	Test mode description
Stand-alone Self-Shmoo (Through Komodo I2C)	Once a system has started, the system can boot into an alternate configuration by: 1. Software running on the DUT edits configuration registers via IFC 2. Switch sampling is disabled (RCFG_CTL[SAM] = 0). 3. Reset restart is asserted (RCFG_CTL[GO] = 1). - DIP switches are NOT used to set configuration registers. - Configuration registers drive pins statically and/or dynamically during the reset configuration sample window, as appropriate. 4. System released from reset.
Through Komodo I2C configuration	At power up through the switch sw_komodo_config_sel, the komodo card can be used to load the configuration rather than loading it from the DIP CFG switches

2.4 Lead-Free/RoHS

All components are lead-free/RoHS compliant.

Chapter 2
Architecture

The **PSC9132 QDS** architecture is primarily determined by the BSC9132 processor, and by the need to evaluate as many of its features as possible, maximizing testability without impacting the ability to deliver an easily usable off-the-shelf software development platform.

Table 2-1 lists the major pin groupings of the BSC9132.

Table 2-1. BSC9132 blocks groupings summary

Signal Group	Details
Demultiplexing	Section 2.1, “Demultiplexing
Memory Controllers	Section 2.2, “DDR”
SerDes Ports	Section 2.3, “SerDes ports”
SerDes - CPRI	Section 2.3.1, “CPRI support”
SerDes - SGMII	Section 2.3.2, “SGMII support”
SerDes - PCIe	Section 2.3.3, “PCIe support”
IEEE 1588	Section 2.4, “IEEE-1588™ support”
USB	Section 2.5, “USB interface”
IFC Bus	Section 2.6, “IFC bus”
I2C	Section 2.7, “I2C”
SPI	Section 2.8, “SPI interface”
Interrupts	Section 2.9, “Interrupt controller”
UART	Section 2.10, “Serial ports”
ADI	Section 2.11, “ADI interface”
GPS	Section 2.12, “GPS”
SDHC	Section 2.13, “SDHC interface”
USIM	Section 2.14, “USIM interface”
TDM	Section 2.15, “TDM interface”
Debug Support	Section 2.16, “Debug support”
JTAG	Section 2.16.1, “JTAG port”
GPIO	Section 2.17, “GPIO controller port”
DMA	Section 2.18, “DMA controller”
Thermal	Section 2.19, “Temperature monitoring”

Table 2-1. BSC9132 blocks groupings summary (continued)

Signal Group	Details
Reset	Section 2.20, “Reset”
Clock	Section 2.21, “Clock”
Power	Section 2.22, “Power”
Qixis FPGA Architecture	Section 2.22, “Power”
Qixis Program Model	Section 2.22, “Power”
Config	Section 2.22, “Power”
Appendix	Section 2.22, “Power”

2.1 Demultiplexing

The BSC9132 supports a large number of internal configuration options, so the first level of architecture for the device involves using external demultiplexers to route internal BSC9132 QDS logic to suitable external logic. As an example, on the BSC9132, certain UART pins may be connected to either a RS-232 PHY or a USB transceiver, but not both at the same time. An array of configurable multiplexers surrounding the device routes the UART/USB signals to an RS-232 PHY or a USB PHY, under software control.

It is useful to consider the multiplexers as converting the device into a “virtual large-pinned” device; however, it is important to remember that even with that model, not all of the interfaces can be used at the same time.

Note that this flexibility means that software configuration plays a large part in the decision about which peripherals are present and which are not. Software should examine both the configuration pin inputs and user-specified settings, to determine how to properly configure a system.

For further information on board configuration, refer to the respective block sections (USB, UART, SDHC, and other related sections) and the general configuration chapter.

Table 2-2. BSC9132 QDS demultiplexing configuration

Block	SW: BRDCFGx Field	HW: Demux Signals	cfg	Connects To	Description
SPI1	BRDCFG3[0]	FPGA_SPI1_SEL_L	0	DUT_SPI	On-board SPI1 bus
			1	UART3	On-board PHY
	BRDCFG3[1]	SPI1_FLASH_SEL_B	0	MUX_SPI	On-board SPI1 devices
			1		1588 IF connecter SPI1_SLICMUX
	BRDCFG3[2]	SLIC_SPI1_SEL	0	SPI1_SLIC_MUX	On-board SLIC PHY1
			0		On-board SLIC PHY2

Table 2-2. BSC9132 QDS demultiplexing configuration (continued)

Block	SW: BRDCFGx Field	HW: Demux Signals	cfg	Connects To	Description
SDHC	BRDCFG3[2]	SDHC_MUX_SEL_L	0	DUT_SDHC	On-board SD Card
			1	DUT_USIM	On-board SIM Card
SerDes	BRDCFG4[0:3]	SEL[A,B,C,D]	0011	PEX2,CPRI[2:1]	PCIe slot; SFP+[2:1]
			0001	PEXx2, SGMII1,CPRI1	PCIe slot; Eth Por P5 SFP+[1] J42
			0000	PEX2,SGMII[1:2]	PCIe slot; Eth Port P5,P6
			0111	PEX, SGMII[2] CPRI[2:1]	PCIe slot; Eth Port P6 SFP+[2:1] (J[43:42])
			0101	PEX, SGMII[2:1] CPRI[1]	PCIe slot; Eth Port P[5:6], SFP+[1] J42
			1111	SGMII[1:2] CPRI2:1]	Eth Port P[5:6], SFP+[2:1] (J[43:42])
USB/I2C2/ UART2	BRDCFG5[0:1]	FPGA_USB_MUX[0:1	00	Disconnection	
			01	DUT_ULPI	USB PHY
			10	GPIO	to/from FPGA
			11	UART2,I2C	UART PHY, I2C switch
OCM_MUX	BRDCFG6[0]	CFG_OCM_UART	0	DUT UART1	PHY COM2
			1	OCM UART	
GPS_MUX	BRDCFG6[1]	CFG_GPS_UART	0	UART3	Header J24
			1		GPS Module U136
UART0_MUX	BRDCFG6[2]	UART_MUX_SEL_L	0	UART0	PHY COM1
			1	DUT GPIO	to/from FPGA
ANT12CK_ MUX	BRDCFG6[6:7]	VCVR_REF_SEL[0:1]	1x	RF1 REF_CLK	DUT ANT1_REF_CLK
			11	RF3 REF_CLK	
ANT12CK_ MUX	BRDCFG6[6:7]	VCVR_REF_SEL[0:1]	0x	RF1 REF_CLK	DUT ANT1_REF_CLK
			1x	RF3 REF_CLK_3	
			x0	RF2 REF_CLK	DUT ANT2_REF_CLK
			01	RF3 REF_CLK_4	

2.2 DDR

The BSC9132 includes two DDR Controllers - DDR_1 controller referring to two e500 processors, and the second one to two StarCore SC3850 DSP subsystems.

The BSC9132 QDS supports two on-board DDR3 memory banks with industry-standard DDR3 memory devices as 4x4Gb DDR3 memories (MT41J512M8THD-187E) from Micron. Each of memory banks has dedicated power supply that provides necessary GVDD, VTT, and VREF voltage rails with both 1.35V/1.5V voltage versions.

The BSC9132 DDR controller, GVDD power plane is separated from DDR devices power voltage planes for the DUT DDR controller current monitoring. The memory interface includes all necessary termination and I/O power, and is routed with GND reference plane with specify trace matching so as to achieve maximum performance on the memory bus.

The BSC9132 DDR3 controller has 4 chip select signals. Out of four, only chip selects 0 and 1 are used by default and other chip selects are selected by resistor mounting options on a 3-pad resistor. The table below shows the mounting options.

Table 2-3. DDR MCS[0:3] connection options

Signal	Resistors mounting option	
1	Install	Remove
D1/D2_MCS0_B	R585-A/R714-A	R587-B/R714-B
D1/D2_MCS1_B	R587-A/R717-A	R587-B/R717-B
D1/D2_MCS2_B	R587-B/R714-B	R585-A/R714-A
D1/D2_MCS3_B	R585-B/R717-B	R587-A/R717-A

The DDR memory architecture for BSC9132 QDS is shown in Figure 2-1.

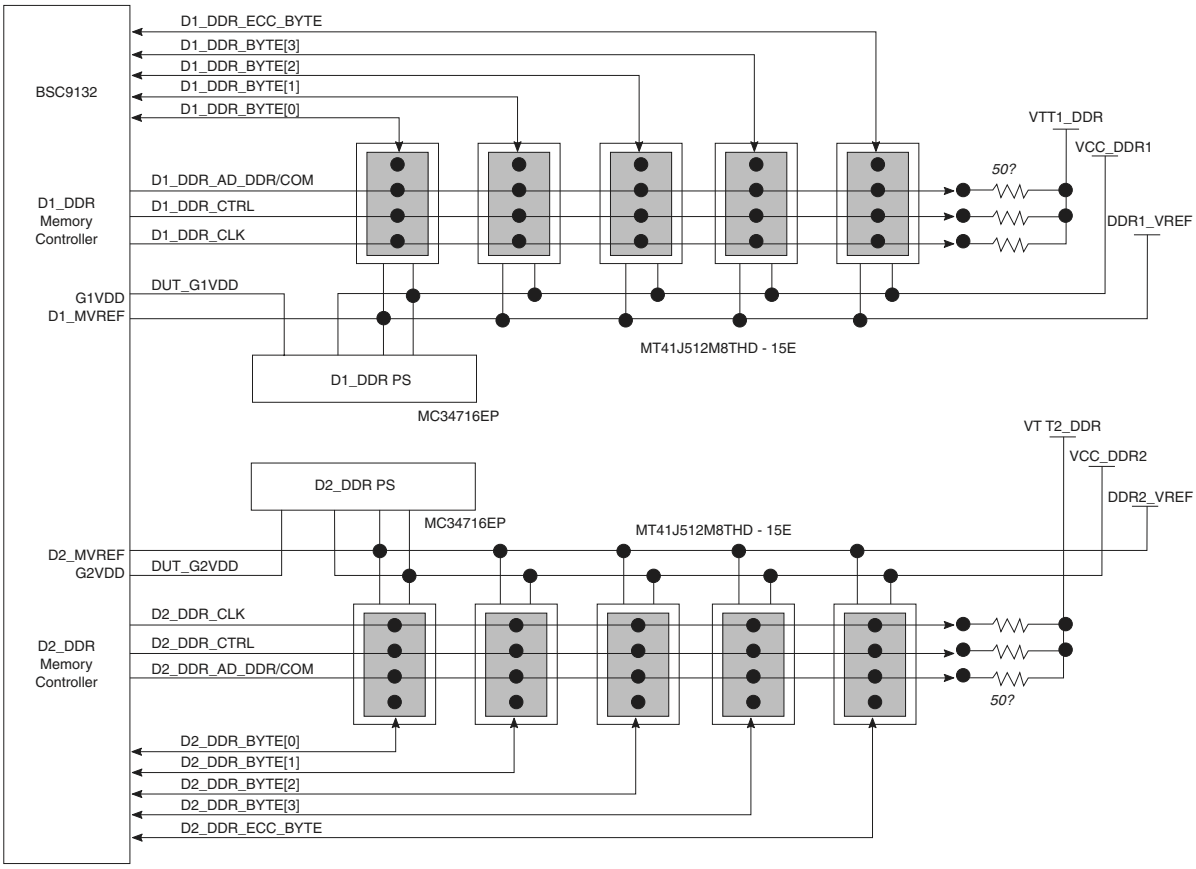


Figure 2-1. BSC9132 QDS memory architecture

2.2.1 Compatible DDR3 devices

The DDR interface of the BSC9132 QDS works with any JEDEC-compliant, 240-pin, DDR3 DIMM module. Table 2-4 shows several DDR3 memory chips that are believed to be compatible.

Table 2-4. DDR3 memory device

Mfg.	Part Number	Size	Ranks	ECC	Data Rate	Verified?
Micron	MT41J512M8THD-15E[1]	2 GB	2	Y	1060	Yes
Micron	MT41J512M8THD-18E[1]	2 GB	2	Y	1333	Yes

[1] This is an obsolete part. For a redesign or respin, you will need another part from Micron.

2.3 SerDes ports

The SerDes block provides high-speed serial communications interfaces for several internal protocol modules.

The SerDes block provides 4 serial lanes that may be partitioned, under control of the RCW parameters. Note that the term lane is used to describe the minimum number of signals needed to create a bidirectional communications channel. In the case of PCI Express or SGMII, or CPRI a lane consists of two differential pairs, one for receive and one for transmit.

BSC9132 SerDes lanes are grouped two banks each of which receives a dedicated clock: SD1CLK, SD2CLK.

The BSC9132 chip includes various lanes that can be configured to support one or more of the following protocols, given an appropriate clock:

- PEX - PCI Express at 2.5 Gbps or 5 Gbps
- CPRI - Common Public Radio Interface at 6.0 Gbps
- SGMII - Serial GMII (Gigabit Ethernet) at 1.25Gbps or 3.125 Gbps

In order to support the given test cases (Table 3), multiplexers are used to route the SerDes lanes to slots where they are of most use:

- PEX - routed to PCI Express slots as 2xPCIe link.

Note that slots are physically larger (x16 slots) to accommodate larger cards.

- CPRI - routed to SFP + connecter to accommodate optical transceiver.
- SGMII - routed to a SGMII PHY (VSC8221) with standard Ethernet RJ-45connecter termination.

There are a total of six multiplexers that are controlled by following signals SELA, SELB, SELC, and SELD and a corresponding BRDCFG4 register field (see Section 5.8, “Board Configuration Registers”). The specific combination of these signals can be a selected path as shown in Table 2-5 below.

Table 2-5. BSC9132 QDS supported SerDes lane configuration

Test Case	SerDes-Port/Protocol cfg_io_ports[0:6]	Protocol Muxing				Test Case Selection
	SerDes Number	SD1		SD2		{A,B,C,D}
	Lanes	A	B	C	D	4’bxxxx
0	7’b000_0000	off	off	off	off	4’b0000
22	7’b001_0110	PEX x2		CPRI#2	CPRI#1	4’b0011
27	7’b001_1011	PEX x2		SGMII#1	CPRI#1	4’b0001
32	7’b010_0000	PEX x2		SGMII#1	SGMII#2	4’b0000
33	7’b010_0001	PEX x1	SGMII#2	CPRI#2	CPRI#1	4’b0111
38	7’b010_0110	PEX x1	SGMII#2	SGMII#1	CPRI#1	4’b0101
43	7’b010_1011	SGMII#1	SGMII#2	CPRI#2	CPRI#1	4’b1111

NOTE

The term lane is used to describe the minimum number of signals needed to create a bidirectional communications channel; in the case of PCI Express or SGMII and CPRI, a lane consists of two differential pairs, one for receive and one for transmit, or four in all.

Figure 2-2 shows an overview of the SerDes routing, including the multiplexing required.

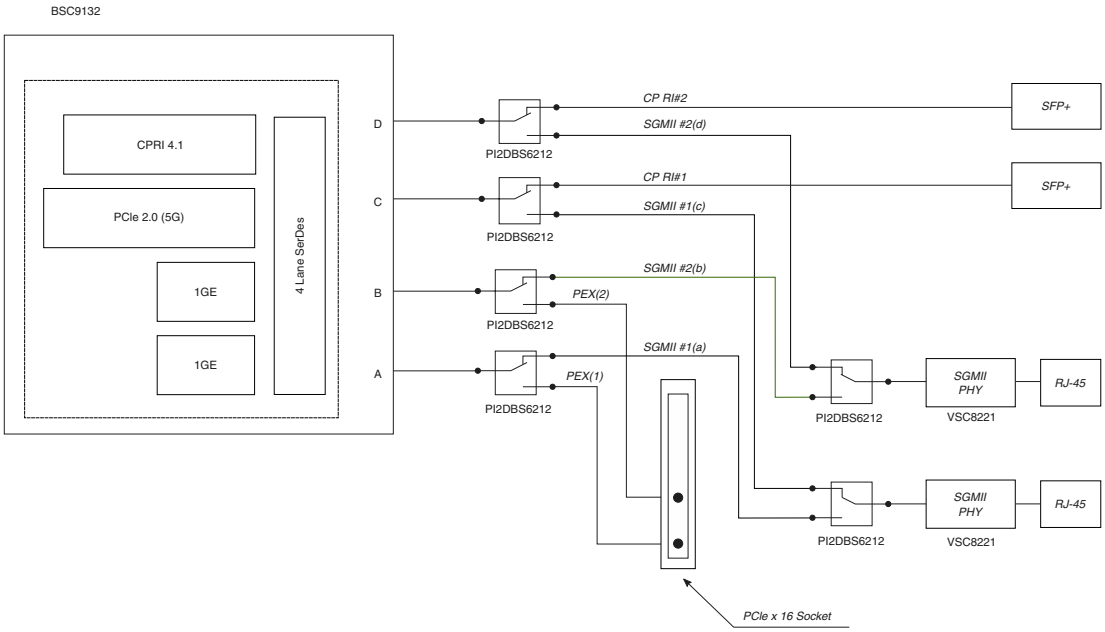


Figure 2-2. SerDes routing

2.3.1 CPRI support

The BSC9132 QDS supports evaluation of two CPRI links protocol over the SerDes lanes. CPRI protocol terminated onboard SFP+ carriages that allow to plug in optical transceiver such as FTLF8528P2BCV from Finisar. The recommended SFP+ devices part number are shown in Table 2-6.

Table 2-6. Hot-pluggable SFP+ devices

Manufacturer	Product Number	Reference
Finisar	FTLF8528P[1]3BxV	8.5 Gb/s Short-Wavelength SFP+ Transceiver Tri-Rate 2.125/4.25/8.5 Gb/s Fibre

[1] Built-in digital diagnostic functions; Single 3.3V power supply

Except for special test purposes, only the above settings should be used.

Note that CPRI on chip controller used with SD2_REFCLK only should be set properly for the required data rate (122MHz, by default).

For the CPRI sharing mode supporting (CPRI1 and CPRI2 lane works in sharing mode) BSC9132 has specific CP interface signal's set. BSC9132 QDS boards has a provision for the CP_RXCLK external output and SD2_REFCLK external input for the Sd2_REFCLK loop closing over the jitter cleaning device.

2.3.2 SGMII support

The BSC9132 QDS supports up to two 10/100/1000baseT triple-speed Ethernet Controllers (TSEC). Each TSEC port can independently be selected to specify speed of SGMII over the SerDes protocols. This is achieved by using single VSC8221 10/100/1000BASE-T SGMII PHY with 1.25 Gbps SerDes from Vitesse. Each PHY on board gets dedicated 25 MHz clock and used internal PLL for clock multiplication. There is available one common 1.2V LDO for both PHYs core voltage supply. Both Ethernet ports use MII link for PHYs configuration issue. Power up configuration performs by voltage setting at the specific PHY's CMODE[0:3] inputs by wire strapping. Each of the two ethernet ports terminated by magnetic with standard RJ-45 shielded connecter that allows to use standard CAT-5a type of copper ethernet cable.

Table 2-7. BSC9132 QDS ethernet port configuration

EC #	Supported Protocols	Configuration Required	Interface Voltage	PHY Address	Connector Location	Status Indicator
1	MII, SGMII	CMODE0[3:0] = 0000 CMODE1[3:0] = 1010 CMODE2[3:0] = 1111 CMODE3[3:0] = 0000	LVDD (2.5V)	0	P5 RJ45 connecter.	L(yellow): 100 linked R(green): 1000 linked
2	MII, SGMII	CMODE0[3:0] = 0001 CMODE1[3:0] = 1010 CMODE2[3:0] = 1111 CMODE3[3:0] = 0000	LVDD (2.5V)	1	P6 RJ45 connecter.	L(yellow): 100 linked R(green): 1000 linked

2.3.3 PCIe support

The BSC9132 QDS supports evaluation of the PCI express protocol using the existing PCI Express slots.

By default, board and chip are configured as Root Complex PCIe device, when REFCLK is provided by on board PLL (100 Mhz clock) to DUT SD1_REFCLK input and to PCIe slot. There are optional SD1_REFCLK mux that allows to use board as PCIe EndPoint with external reference clock from PCIe slot. To achieve this option, you need to set rst_force3[2] FPGA register bit to1'b1 and plug PCIe card onto the board slot. This option is used during chip validation flow when BSC9132 QDS is linked through PCI Express cable (see Figure 2-3 below).

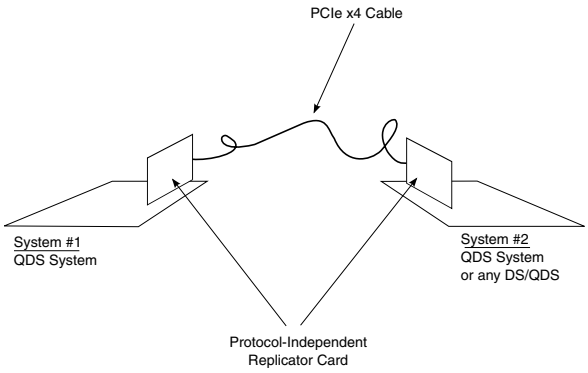


Figure 2-3. PCI Express board to board link

In addition, a standard PCI Express cable is needed. They are typically available in 1 - 5 meter lengths, allowing ample room for testing setups.

A listing of tested-compatible SerDes repeater cards is provided in Table 2-7.

Table 2-8. SerDes repeater card listing

Manufacturer	Product Number	Reference
One-Stop Systems Inc.[1]	OSS-PCIe-HIB25-x4-H	http://www.onestopsystems.com/PCIe-HIB25-x4.php

[1] Use only the “host” mode card; cards designed for installation in target systems typically support driving RESET to the remote system, which is neither needed nor supported on QDS systems. These boards and cables are not available through Freescale.

2.4 IEEE-1588™ support

The BSC9132 QDS includes support for the IEEE 1588 precision time protocol (PTP). This facility works in tandem with the internal ethernet controllers to time-stamp incoming packets. This is supported at a basic level with internal logic and the use of a precision 125.00 MHz reference clock, accurate to ±25 ppm.

In addition, the BSC9132 QDS supports the installation of a special IEEE1588 support card, which monitors ethernet data (including the IEEE-1588 sideband signals) to dynamically alter the 125.00 MHz reference clock to maintain synchronization at a high level. This card is manufactured by third parties, such as Symmetricon Inc., and is not included with the BSC9132 QDS.

Figure 2-4 shows an overview of the IEEE 1588 block.

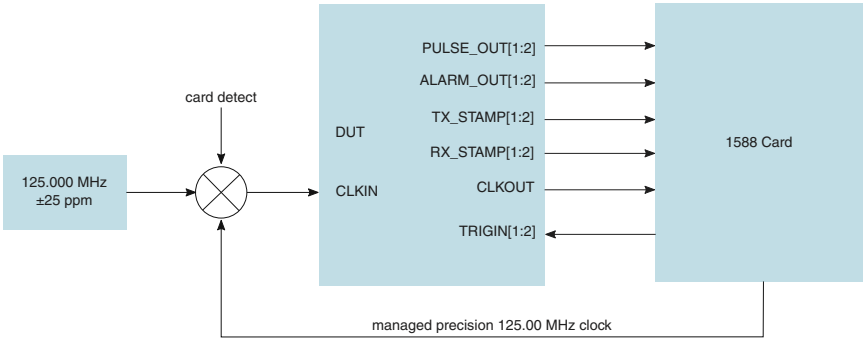


Figure 2-4. IEEE-1588 interface overview

2.5 USB interface

The BSC9132 QDS has one USB 2.0 port that uses the ULPI protocol to external USB PHY connection. The controller may be configured for host or device modes. To allow maximum flexibility for configuration, the BSC9132 QDS multiplexes the USB port pins with either CVDD-powered I2C port 2 signals or various CVDD-powered signals such as FA,TIMER, and GPIOs. The BSC9132 QDS has an on-board demultiplexer in order to select required used interface controlled by QIXIS FPGA.

To handle the differing voltage rails this presents, SMSC USB PHY USB3315 is selected, which supports various CVDD voltages without using voltage level translators. The choice of USB port to enable, is typically guided by requirements as to which signals must be available. Based on the register settings, USB ports are available. Refer to the ULPI Bus register settings, for FPGA Programming registers.

Table 2-9 below includes the overall USB signals multiplexing the BSC9132 QDS, while Figure 2-5 shows a logical breakdown of the USB components.

Table 2-9. USB Mux options

FPGA signal	Value	Selected port/signals
FPGA_USB_MUX[1:0]	2'b00	USB (default)
	2'b01	GPIO[0:3] UART2 I2C2
	2'b00	GPIO[69:72],GPIO[62:63] TIMER1 TRIG_IN

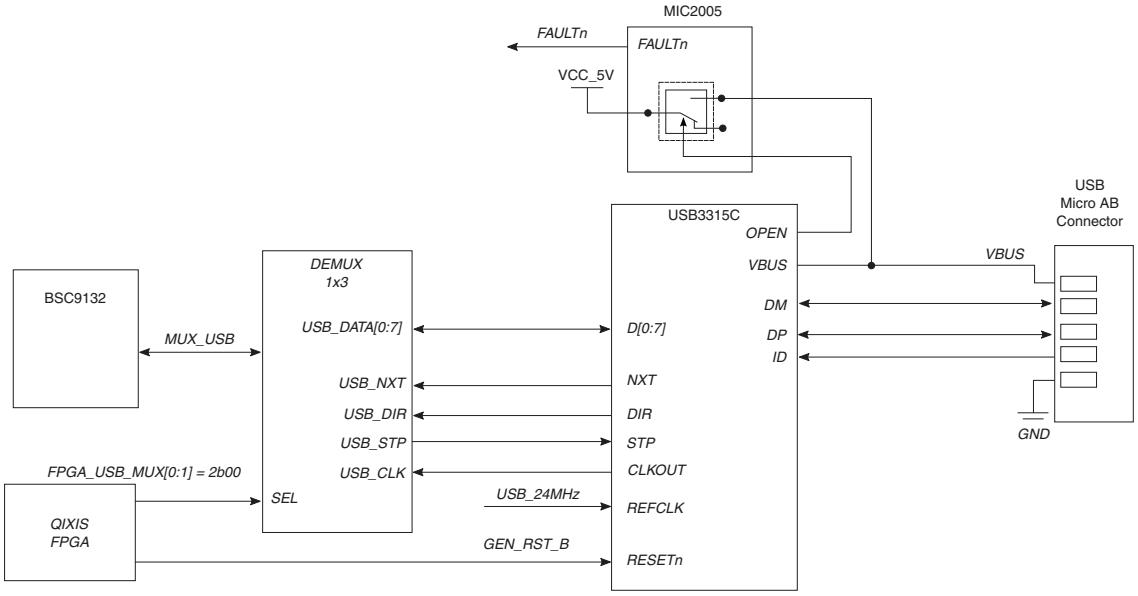


Figure 2-5. USB architecture

2.6 IFC bus

BSC9132 supports Integrated Flash Controller that enable the SoC to access NAND Flash, NOR Flash, SRAM, and Generic ASIC memories.

Features supported are

- 16-bit multiplexed Address/data bus
- x8/x16 NAND devices
- Support for 256 MB NOR devices (x8/x16)
- 3 chip selects
- One IFC output clock
- ECC generation and checking for NAND

On the BSC9132 QDS, this includes:

- socketed card for high-speed device (SRAM, ASIC) evaluation
- socketed card for testport use
- evaluation of BVDD operating ranges
- socketed NAND flash (8-bit, large or small page support)
- socketed NOR flash (16-bit)
- NAND flash 8-bit
- NOR flash 16-bit

- virtual bank support for NOR
- connecter for PromJet (NOR flash emulator)
- QIXIS registers
- QIXIS RCW memory.

NOTE

PromJet modules are flash memory emulators available from Emutec (www.emutec.com). DS systems use the 16-bit wide devices (size is user-dependant), with the *low-voltage* option for use on the 3.3V local bus.

To effectively manage all these resources, with the maximum amount of performance and flexibility, the BSC9132 QDS divides the IFC into two segments: a high-speed and low-speed local bus.

Figure 2-6 shows an overview of the segmented IFC bus.

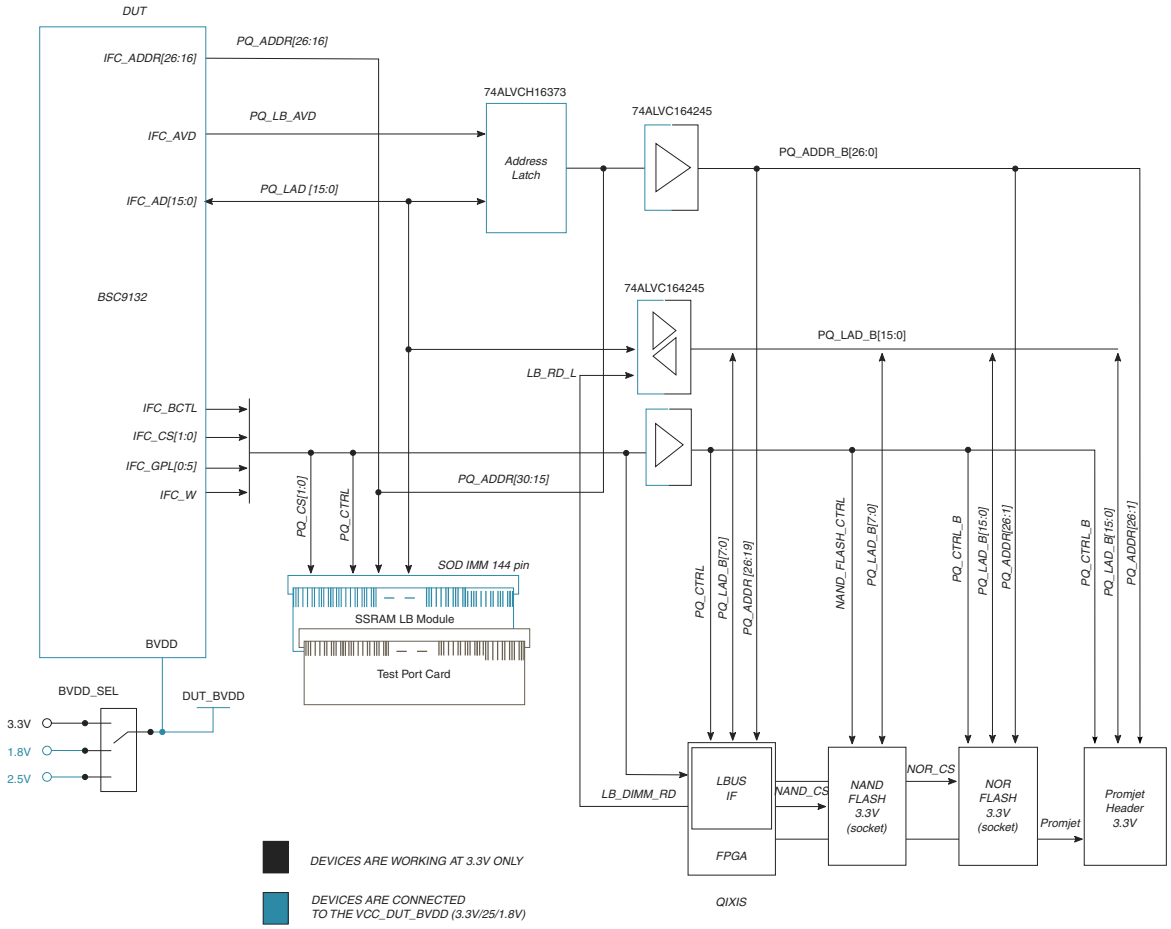


Figure 2-6. BSC9132 QDS segmented IFC bus architecture

2.6.1 High-speed IFC bus

The BSC9132 QDS high-speed IFC bus is shown in Figure 2-6. It is essentially a direct connection to the entire IFC port. The IFC/Testport card is responsible for demultiplexing the LAD bus, if needed, and for performing any other actions the card requires.

The only unique requirement is that the IFC/Testport card must generate the LB_CTL_HS signal; this is an active-low signal used to indicate the low-speed bus side (through **QIXIS**, which manages it) that the high-speed side is handling the IFCtransaction. The signal is pulled up, so if no HSLB/Testport card is installed, the low-speed side handles all transactions, as expected.

2.6.2 Low-speed local bus

The low-speed side of the local bus is active whenever the high-speed side has not claimed a transaction as described previously. For the low-speed side, the multiplexed address/data signals (LAD[0:26]) are latched, and the resulting address, 16 bits of data, and all other IFC signals are buffered and translated to 3.3V. At this point, the local bus is virtually identical to existing development systems, such as the P4080DS.

Since the chip-select signals flow through **QIXIS**, several flexibility options for local-bus devices become possible:

- Dynamic reassignment of LCS0_B (the IFC boot device) to NOR, NAND, or PromJET.
- Treating large NOR devices as an array of smaller flash devices (“virtual bank”).

IFCl bus chip select mapping on the low-speed side is summarized in Table 2-10. The “cfg_lbmap” setting, as defined by the BRDCFG0 configuration value (see Section 5.8.1), is used to remap the chip-select signals.

Table 2-10. Local bus chip select mapping

QDS Manual SW cfg_lbmap[1:3]	BRDCFG0 cfg_lbmap[0:2]	NOR Flash	NAND Flash	PromJet	SRAM DIMM	QIXIS
000	000	LCS0	LCS1	-	-	LCS2
100(default)	100	LCS1	LCS0	-		
111	111	LCS2	LCS1	LCS0-		
101	101	-	LCS0	LCS1	LCS1	

Table 2-11. IFC bus on board devices

Device Type	Schematic RefDes	Part Number	Package	Remarks
NandFlash 8-bit	U50	K9F1G08U0C	TSOP48	Socket
NorFlash 16-bit	U38	JS28F512M29EWL	TSOP56	Socket
NandFlash 16-bit	U157	MT29F2G16AADWP	TSOP48	PS-PCB
NorFlash 8-bit	U158	JS28F512M29EWL-	LCS0	LCS1

Device Type	Schematic RefDes	Part Number	Package	Remarks
PromJet	J19	--	-	External
QIXIS FPGA	U19	A3PE1500 Actel	FGG676	8-bit bus

2.6.3 Virtual bank

The virtual bank feature is available only for NOR flash, and only when it is selected as the device connected to LCS0_B. In that case, the value of cfg_lbmap[1:3] is driven into three XOR gates, which toggle the MSB's of the NOR address, as shown in Figure 2-7.

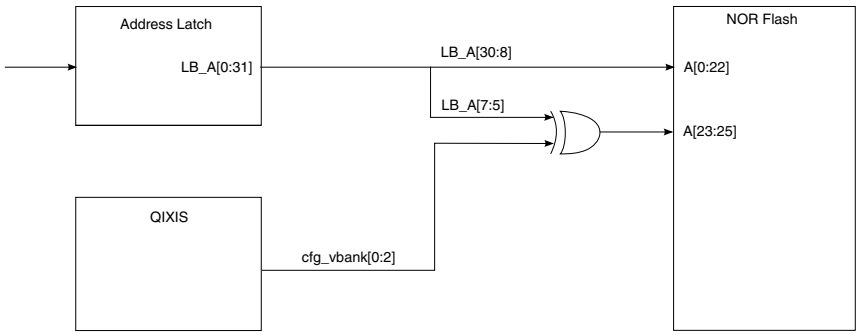


Figure 2-7. NORFlash virtual bank address XOR

When LBMAP=0000 and VBANK=000 so LB_A[5:7] is not altered, and the NOR flash behaves normally.

If LBMAP=0100 and VBANK=100, LB_A[5] is toggled effectively swapping the top and bottom halves of the NOR flash. If program "A" was stored in the bottom half, and program "B" in the top half, setting LBMAP swaps the two program images. This allows easy evaluation of boot code, without losing known-good images, using the following sequence:

1. with LBMAP=0000 (default), boots known-good image in "A"
2. program test image to area "B"
3. reconfigure and reboot with LBMAP=0100, system boots from "B"
4. if image is good, set LBMAP=0000 and overwrite image "A"
5. if image is bad, press reset or wait for watchdog-system resets with LBMAP=0000 and boots known-good image in "A".

Similarly, using more significant digits of VBANK can create 4 (LBMAP=0XX0) or 8 (LBMAP=0XXX) virtual banks, allowing multiple bootable images. Note that each VBANK reduces the overall maximum size.

Table 2-12. IFC bus chip select mapping

NOR Zones (1/8 of 128MB)	VBANK			
	000	001	010	011
A	A	B	C	D
B	B	A	D	C
C	C	D	A	B
D	D	C	B	A
E	E	F	G	H
F	F	E	H	G
G	G	H	E	F
H	H	G	F	E

In the above table, the NOR is partitioned into eight 16MB “zones”, which can be arranged under control of VBANK. Note that incrementing VBANK has the effect of moving each “zone” into the first position, allowing easy sequencing of various images.

2.7 I^2C

The QDS system architecture is based on remote configuration of the system over I^2C; for this reason, boards have a much larger array of I^2C present as compared to the existing DS systems. Even though the BSC9132 QDS supports up to two I^2C buses, in order to make the I^2C resources available to both local and remote systems, the QDS systems attach all boot-software-dependant devices and DDR3 SPD EEPROM to the I2C1 port, with the remainder being used for chip validation procedures like DUT power voltage/current measurement and test environment control, or used for board-to-board communication.

Each DUT I2C channel is connected to on-board I2C devices network through fan-out expansion I2C bus device MAX3394. This device includes additional features as voltage translation and tri-state output mode that allows to use it as an I2C bus isolator. These options are used on the BSC9132 QDS board to switch between I2C masters of a specific I2C channel. So I2C1 or I2C2 channel peripheral devices can be controlled either by Komodo I2C master or by DUT. With all devices attached to I2C2, multiplexers are used to manage the heavy loading that would otherwise be present, which partitions the I^2C bus into several sub-buses by a 4-channel I2C bus switch, PCA9546A. The SCL/SDA upstream pair fans out to four downstream pairs, or channels. Any individual channel or combination of channels can be selected, determined by the contents of the programmable control register. This device has its own system I2C address. Four downstream channel devices can use identical I2C address as well. Figure 2-8 shows the BSC9132 QDS I2C port.

Table 2-13. BSC9132 QDS I2C port 1 - devices address list

DUT I2C Port	I2C Bus segment	I2C Device Address (7'bxxx_xxxx	Device purpose	Device	Notes
1	PROC_ISO_I2C1	~	I2C Master/Slave	BSCBSC9132	
	I2C1	0x50	RCW+PBL	MEM EEPROM 8K AT24C64C	Stores RCW and PBLOADER data. Write protectable
		0x57	System ID	MEM EEPROM 256K AT24C02C	Stores board specific data, including MAC addresses, serial number/errata and other related information. Write protectable.
		0x77	SPD Data	MEM EEPROM 256K AT24C02C	DDR_D1 Memory Bank SPD. Contents follow established SPD standards
		0x79	SPD Data	MEM EEPROM 256K AT24C02C	DDR_D2 Memory Bank SPD. Contents follow established SPD standards
		0x68	RTC	DS3232	Time and periodic interrupt.
		0x21	Core Power Supply	ZL6100 Power Supply (PMBus)	Controls and monitors VDDC\VDD rail.
	ISO_I2C1	0x55	QIXIS config data	MEM EEPROM 8K AT24C64C	Stores QIXIS configuration data. Accessible while board is powered off. Write protectable.
		0x56	QIXIS Program image	MEM EEPROM 256K AT24C02C	Stores QIXIS GMSA program code. Accessible while board is powered off. Write protectable.
	I2C_KOMODO	0x66	QIXIS BCSR	I2C Slave	Remote control input
		~	Komode/Remote System	I2C Master	May or may not be present, and may or may nor present an I2C address bus discovery operations.

Table 2-14. BSC9132 QDS I2C port 2 - devices address list

DUT I2C Port	I2C Bus segment	2C Switch Addr PCA9546 A	I2C Device Address	Device purpose	Device	Notes
2	PROC _ISO_I2C2	~	~	I2C Master/Slave	BSC9132	
	I2C2	0x70	0x50	SFP1	Small Form-factor Pluggable (SFP) fiber optical transceiver	CPRI 1 channel SFP
			0x50	SFP2	Small Form-factor Pluggable (SFP) fiber optical transceiver	CPRI 2channel SFP
			0x55	IEEE1588 Riser	MEM EEPROM 256K AT24C02C	
			~	PEX Slot	I2C/SMBus device	Address, if any at all, depends upon card
		0x71	TBD	RF1 Card	I I2C/SMBus device I	Address, if any at all, depends upon card
			TBD	RF2 Card		
			TBD	RF2 Card		
			0x42	GPS Module	GPS Module I2C (LEA-6T-0)	
		~ ~ 0x72	0x40	VDDC PWR monitor	INA220 CURRENT/POWER MONITOR I	Reports V+I data for VDD_PL rail
			0x41	G1VDD PWR monitor		Reports V+I data for G1VDD rail
			0x44	G2VDD PWR monitor		Reports V+I data for G2VDD_PL rail
			0x45	DUT Temp monitor		Monitors processor thermal diode
			0x4C	DUT Temp monitor	ADT7461 TEMP MONITOR	Monitors processor thermal diode
			0x20	Voltage rail monitor	AD7993 LIN ADC 10BIT 4CH I2C	Reports V data for DUT rails
			0x22	Voltage rail monitor		

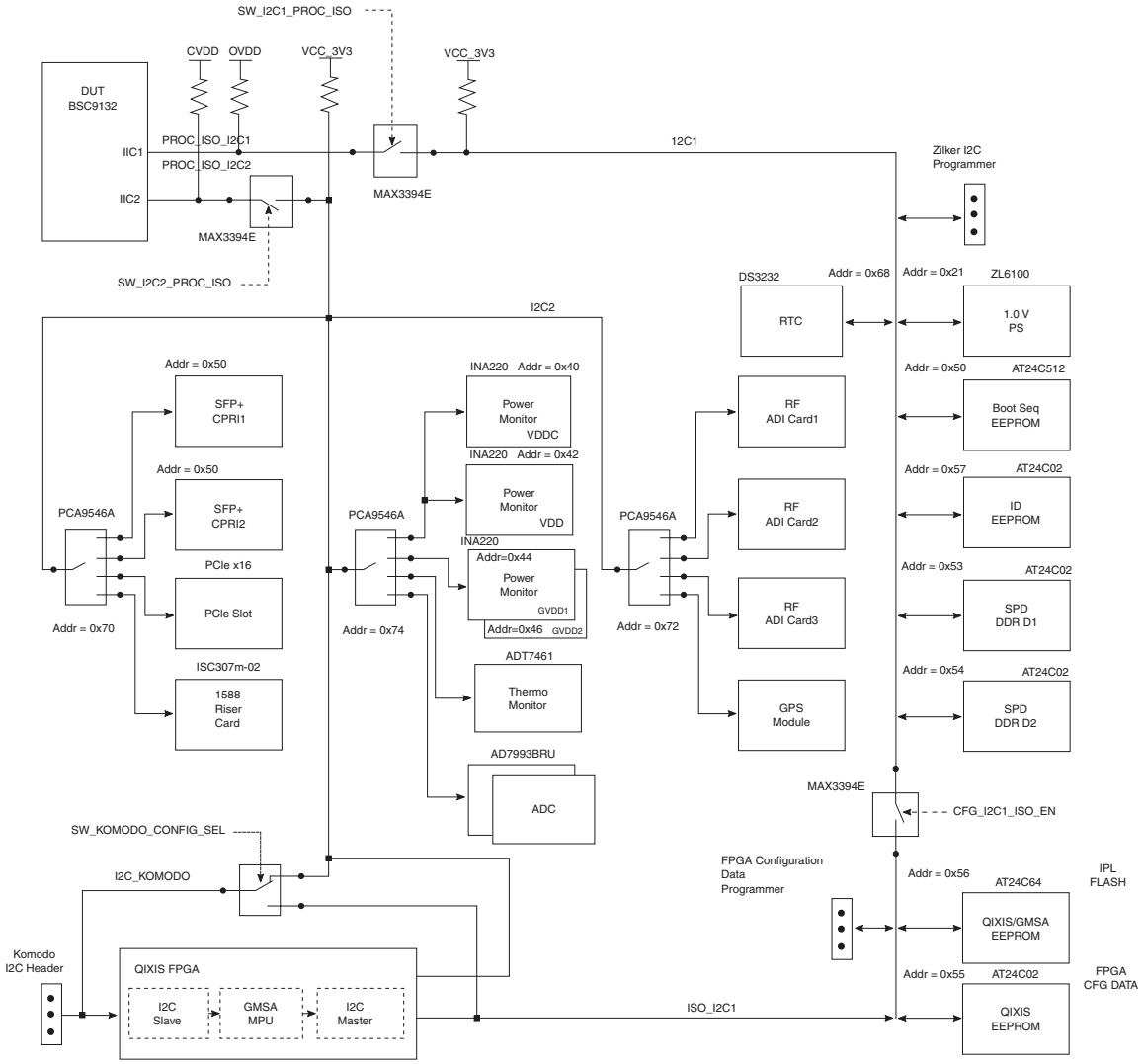


Figure 2-8. 9132 QDS I2C block diagram

An additional difference between the QDS and DS systems is that there may be up to three masters on the I2C bus. In previous systems, it was generally safe to assume there was no other master, and thus no bus retry operations could be needed. Although, in general, the test software plans avoid that tendency, it cannot be assured and so BSC9132 software should support bus error handling.

2.8 SPI interface

The BSC9132 QDS supports two serial peripheral interface (SPI) ports. The BSC9132 SPI is supported with several SPI devices, which operate at a wide variety of sizes, frequencies, and voltages. SPI port 1 has 4 chip select signals and it is connected to various SPI devices via multiplexer. SPI port 2 also has 4

chip selects signals and it is connected to RF connectors. The SPI port 1 of BSC9132 is multiplexed with other on-chip interfaces as UART3, GPIO, and SIM.

Specific interface selection is performed by appropriate setting of chip, PMUXCR3. The default value of this register after chip reset is PMUXCR3[SPI_SIM] = 00 and it allows to use SPI port 1 for boot operation. In order to resolve other interfaces besides SPI1, a multiplexer is used with FPGA_SPI1_SEL_L control signal - see Table for the SPI1 port muxing.

Table 2-15. SPI1 bus muxing table

Mux Name	Signal/Register Value	Destination	Signal/Register Value	Destination
SPI MUX CTRL	FPGA_SPI1_SEL_L = 1'b0		FPGA_SPI1_SEL_L = 1'b0	
DUT PMUXCR	PMUXCR3[SPI_SIM] = 2'b0		PMUXCR3[SPI_SIM] = 2'b0	
DUT Signals	SPI_MOSI	On board SPI memory devices	UART_SIN[3]	On board RS-232 PHY
	SPI_MISO		UART_CTS_B[3]	
	SPI1_CLK		~	
	SPI1_CS0_B		UART_RTS_B[3]	
	SPI1_CS1_B		UART_SOUT[3]	
	SPI1_CS2_B		CKSTP0_OUT_B	QIXIS FPGA
	SPI1_CS3_B		CKSTP1_OUT_B	

The primary device on SPI_CS0_B operates at 3.3V and supplies 32MB, which is sufficient to store uboot, eDINK, or other code for SPI-boot evaluation. Figure 2-9 shows the overall connections of the SPI port1 portion. SPI port 1 is connected to the memory devices EON 32Mb SPI Flash, Winbond's 8Mb Dual read Flash, ST 64Kb SPI Flash, Atmel's 8Mb RapidS Flash, Spansion's 128Mb SPI Flash, 1588 Riser card and SLIC device control interface. Multiplexers are used to manage the heavy trace capacity loading that would otherwise be present because of specific board topology; they partition the SPI 1 bus into two sub-buses. One of them is going to the Flash Memory devices and is selected by the default used signal SPI1_FLASH_SEL_B = 1'b0. When signal SLIC_SPI1_SEL is low, SLIC1 port is selected for the SPI1 port operation.

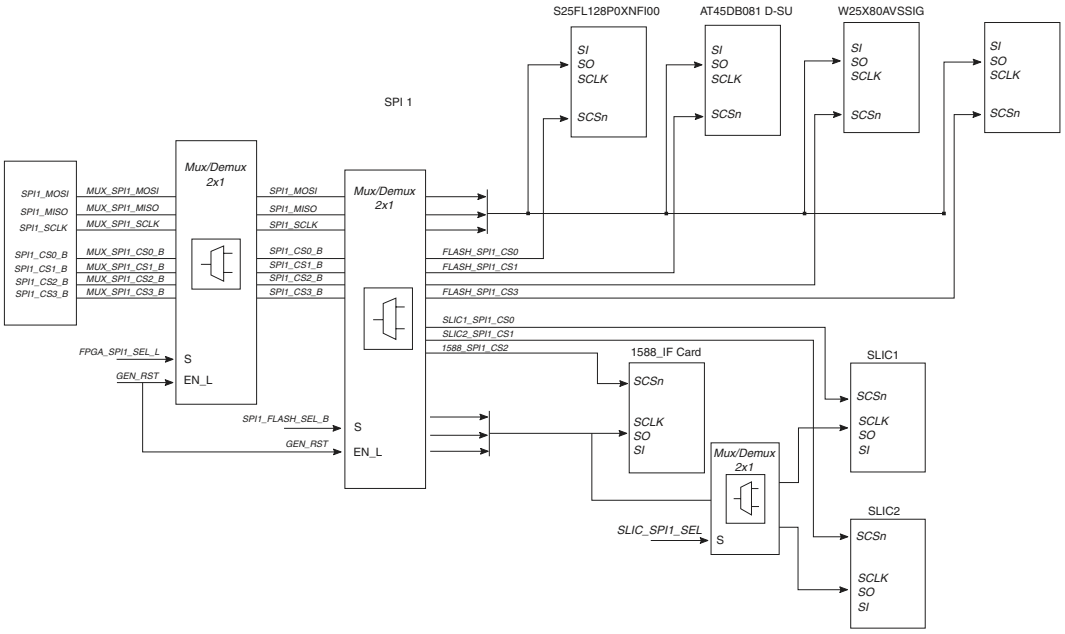


Figure 2-9. BSC9132 QDS SPI1 connectivity

The SPI2 port of BSC9132 resides in the X2VDD power voltage domain and is used for RF cards initialization and control operation. The figure below shows connectivity of the SPI2 port.

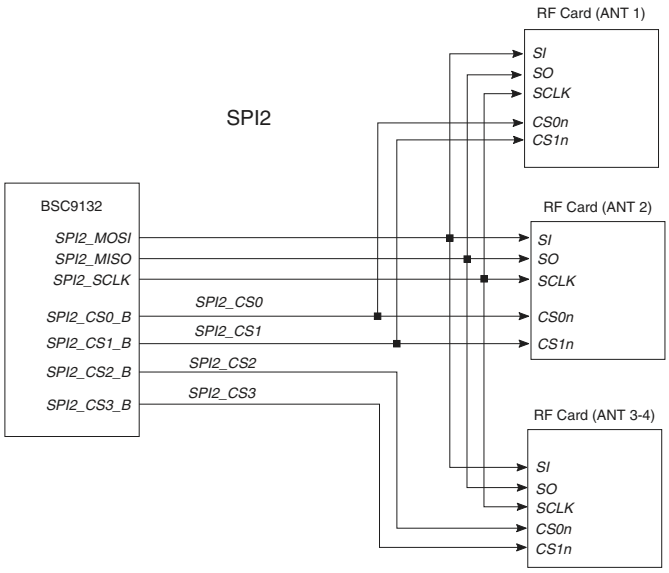


Figure 2-10. BSC9132 QDS SPI2 connectivity

2.9 Interrupt controller

The BSC9132 QDS connects several resources to the BSC9132 interrupt controller, as shown in Table 2-16.

Table 2-16. Interrupt connections [1]

Signal Names	Connections
IRQ0_B	SLIC_INT_B
IRQ1_B	DS3232 Real-time Clock periodic interrupt.
IRQ2_B	AD7998 ADC (VDD_CA, VDD_CB, VDD_PL, GVDD) power supply alert outputs.
IRQ3_B	VSC8221 PHYs 1 interrupts
IRQ4_B	VSC8221 PHYs 2 interrupts
IRQ5_B	USB Power Fault (FLG) for USB ports 1. Indicates over-current at one or more USB connectors.
IRQ6_B	ADT7461 Thermal Alert. Asserted on critical over-temperature fault.
IRQ7_B	Power down event. If enabled (see PX_PWRCTL2), **QIXIS** asserts IRQ7 and delay power-down for approximately 3 seconds.
IRQ_OUT_B	Not used as an interrupt.

[1] The design interrupt sources listed in this table are not connected to the chip interrupt inputs.

NOTE

IRQ[8:11]_B are not available when those pins are used as a 3V-USB port; see Section 2.1, “Demultiplexing” for further details.

2.10 Serial ports

The BSC9132 QDS supports four UART interfaces. Two of them, port UART[0:1] are 4-wire serial-to-serial level transceivers. Port UART[2:3] are reduced 2-wires serial transceivers.

Port 1 and 2 are stacked in dual DB9 male connector placed in the ATX I/O gasket area, so RTS/CTS flow control is supported on these connectors.

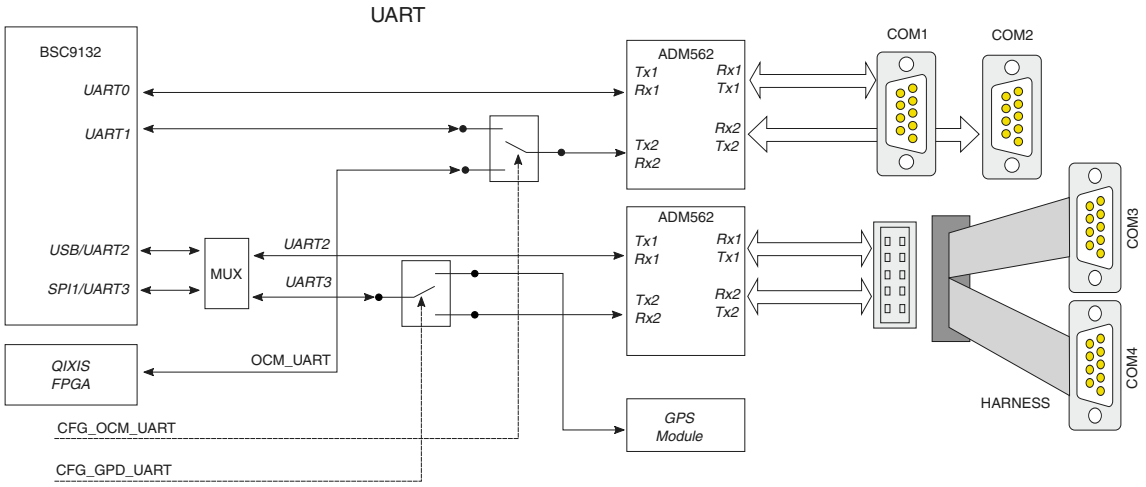


Figure 2-11. UART ports connectivity

All UART ports are multiplexed with other interfaces. Those can be de-multiplexed by configuring QIXIS registers. UART port 3 is connected to the GPS module and also terminated in 2x5. UART 3 is terminated directly to 2x5 Header.

UART1 is not shared with other BSC9132 functions, but is shared with the QIXIS itself; it uses port 1, if permitted by CFG_OCM_UART = 1'b1, to communicate with the user while performing data collections functions, or for configuration purposes. To allow the latter, port 1 is powered from standby power to allow configuration before power is applied.

2.11 ADI interface

The four ANT[1..4] interfaces are supported on the board through the three connectors - RF[1..3] as follows:

- ADI dual port ANT1 interface is connected to the RF1 connecter (X1VDD voltage rail)
- ADI single port ANT4 interface is connected to the RF3 connecter (X1VDD voltage rail)
- ADI dual port ANT2 interface is connected to the RF2 connecter (X2VDD voltage rail)
- ADI single port ANT3 interface is connected to the RF3 connecter (X2VDD voltage rail)

It is expected that with the RF connectors you should use a Xilinx FPGA card so that the data sent from the 9132 AIC block can be tested. This would require an interposer card to connect the QDS and the Xilinx card together. You can perform loopback testing using RF modules.

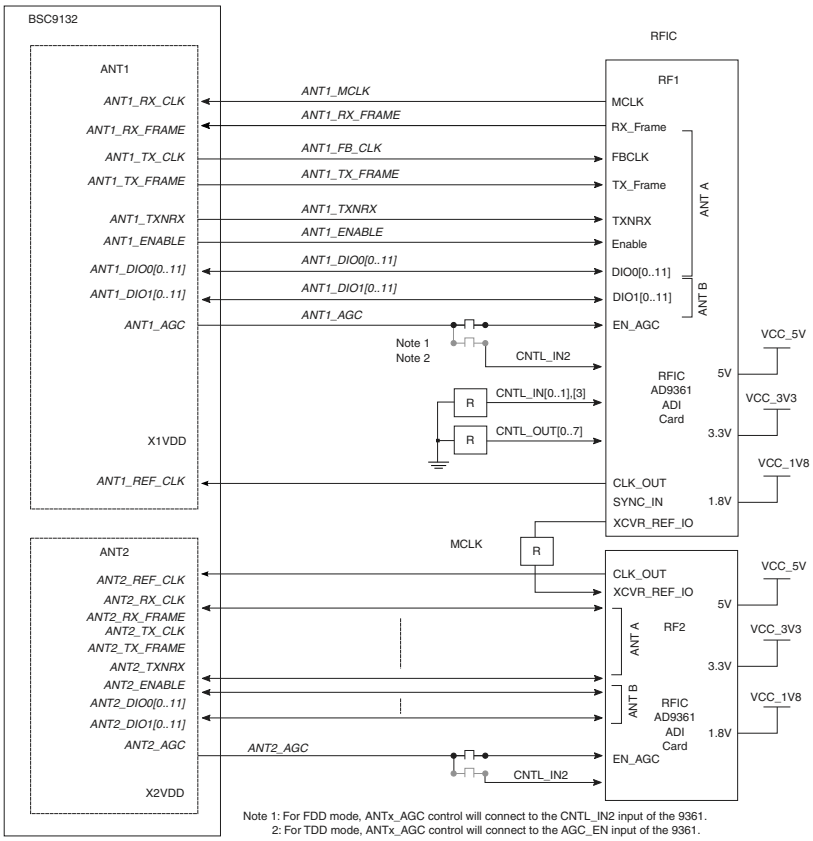


Figure 2-12. BSC9132 QDS ANT1 and ANT2 ADI interface connectivity

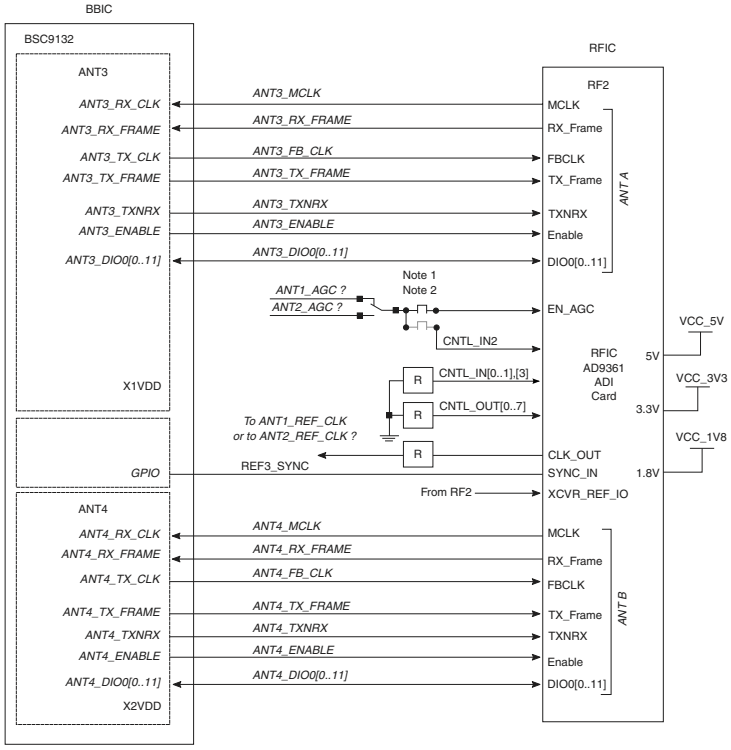


Figure 2-13. BSC9132 ANT3 and ANT4 ADI interface connectivity

Table 2-17. BSC9132 ADI ANT[2:4] interface signals - assembly option

ANT Interface signals	Mux Resisters Install
ANT4_TX_CLK	R964
ANT4_RX_CLK	R996
ANT4_TXNRX	R969
ANT4_ENABLE	R970
ANT4_TX_FRAME	R964
ANT4_RX_FRAME	R967
ANT4_DIO0[0:11]]	R972,R974,R976,R978,R980,R982,R984,R986,R988,R990,R992,R994
ANT3_DIO0[0:11]	R49,R420,R421,R424,R418,R422,R422,R419,R426,R39,R37,R38,R44
ANT2_DIO1[0:9]	R178,R209,R191,R237,229,R233,R235,R220,R24,R183,R200

BSC9132 contains 3 AID interface channels - dual port ANT1 and ANT2 and single port ANT3 and ANT4 respectively. Each channel of chip should get standard 19.2 MHz clock reference as a BBIC device from it's RFIC partner. The BSC9132 QDS board can have 3 RFIC devices - RF1..RF3 for connection. A RF card from Benetel that contains AD9361 high integrated RF agile transceiver offering dual receivers and

transmitters can be used as a RFIC device. Each card includes 19.2MHz clock source is shared between RF cards AD9163 and BSC9132 two AID controllers - ANT1_REFCLK and ANT2_REF_CLK chip inputs respectively. The BSC9132 ANT3 controller input clock is used and ANT1_REFCLK and ANT4 AIC controllers are shared with ANT2_REF_CLK. If we have three RF reference clock sources and only two inputs, we need to leave one of them disconnected. The figure below depicts BSC9132 QDS ANT ref_clock distribution block diagram.

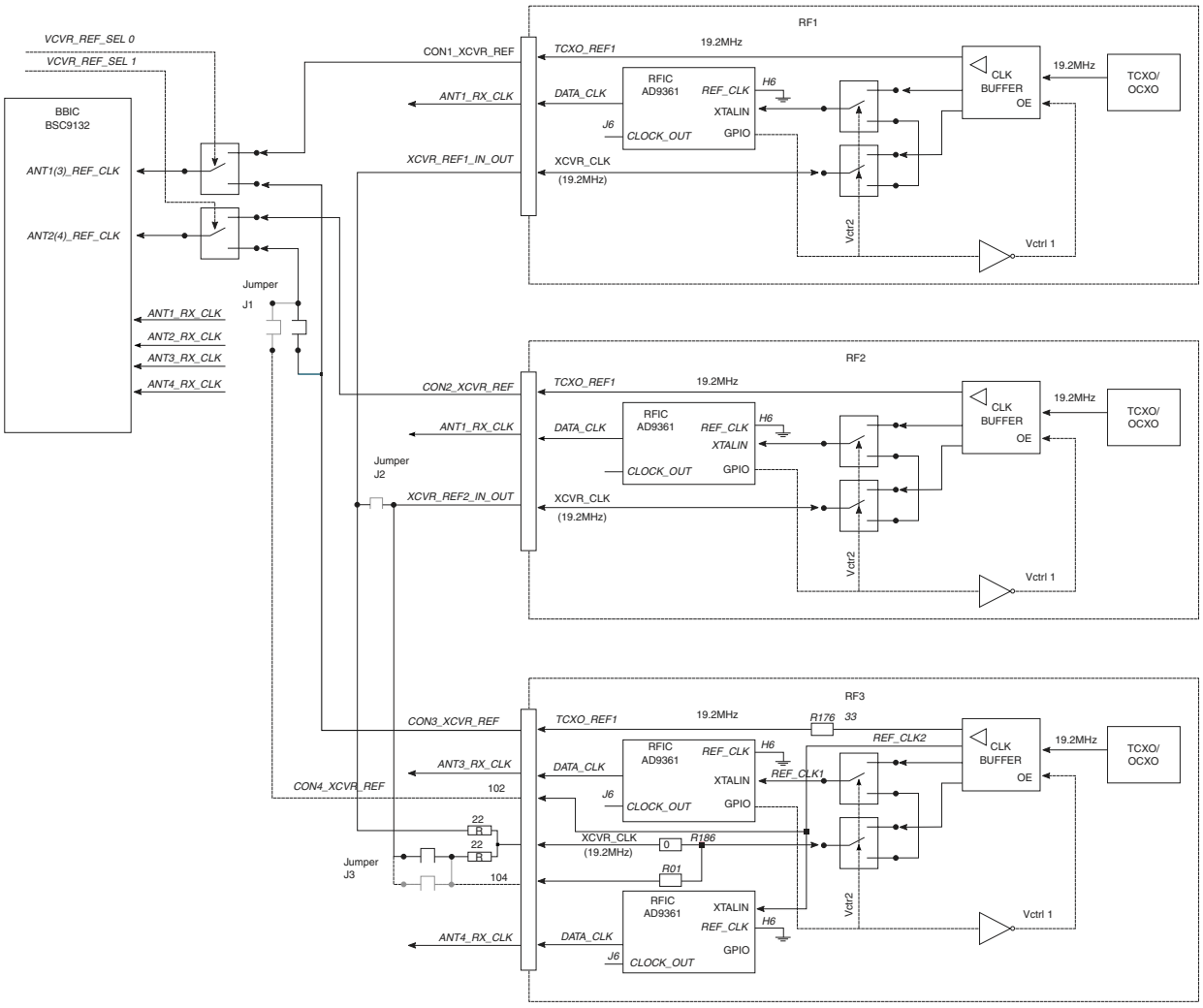


Figure 2-14. BSC9132 QDS AID REF_CLK distribution architecture

2.12 GPS

The BSC9132 QDS supports GPS functionalities. LEA-6T-0 GPS module from uBlox is used in BSC9132 QDS. This module supports the interfaces UART and I2C for remote access. In BSC9132 QDS, UART port 1 and I2C0 channel-6 are connected to the GPS module. Right angle SMA Connector (901-143-6RFX) from AMPHENOL is used as a RF connector for external RF active antenna.

Note that the GPS module hardware settings should be set only for the active antenna. Refer resistor mountings in the UART section for selecting the UART signals for the GPS module.

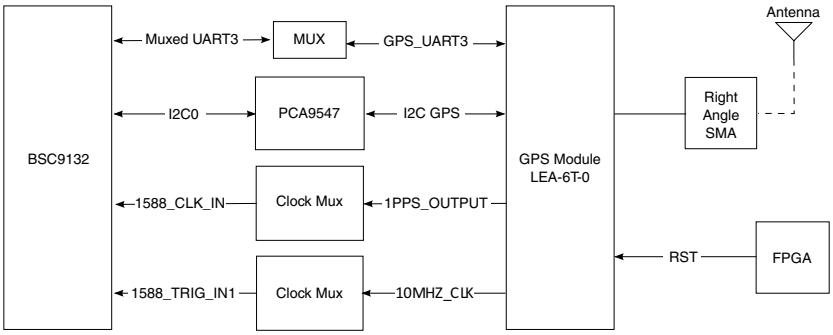


Figure 2-15. GPS module connectivity

2.13 SDHC interface

BSC9132 supports eSDHC module to access SD/SDIO/MMC cards. The BSC9132 QDS provides support for 1-bit/4-bit SD, SDIO, and MMC cards. Data transfer rates supported are:

- MMC: Full Speed (up to 20 MHz), High Speed (up to 52 MHz)
- SD/SDIO: Full Speed (up to 25 MHz), High Speed (up to 50 MHz)

BSC9132 SDHC pins are multiplexed with SIM/TDM/GPIO signals. Based on the below resistor mounting options SDHC interface can be used. The figure below shows the SDHC/USIM interface architecture in the BSC9132 QDS board.

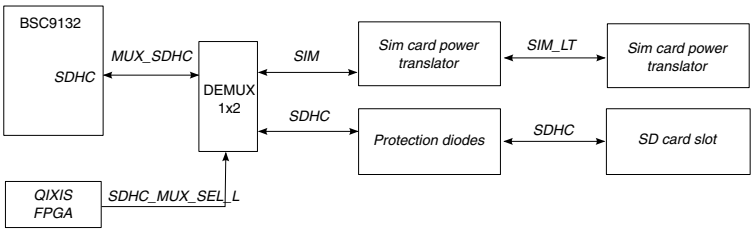


Figure 2-16. BSC9132 QDS SDHC/USIM interface

2.14 USIM interface

BSC9132 supports a USIM interface. USIM is designed to facilitate communication to SIM cards. BSC9132 provides one SIM card interface. Mode of operation supported is:

- Internal one wire interface: In this mode, only TX pin is used to connect the SIM card

The BSC9132 QDS supports CLASS B and C SIM cards. Based on the BVDD voltage settings, SIM_VSEL signal is asserted and the SIM card voltage is selected (Refer BVDD voltage settings for FPGA Programming registers.)

The BSC9132 QDS USIM interface signals are multiplexed with SDHC signals and can be selected by setting the SDHC_MUX_SEL_L FPGA signal (see Table 2-18 below).

Table 2-18. USIM interface signals

Mux control signal	DUT connected signals	SDHC IF signals	USIM IF signals
~	~	1'b0 (default)	1'b1
SDHC_MUX_SEL	MUX_SDHC_CLKI	SDHC_CLKI	SIM_CLK
	MUX_SDHC_CMD	SDHC_CMD	SIM_RST
	MUX_SDHC_DATA0	SDHC_DATA0	SIM_TRXD
	MUX_SDHC_DATA1	SDHC_DATA1	SIM_SVEN
	MUX_SDHC_DATA2	SDHC_DATA2	SIM_PD
	MUX_SDHC_DATA3	SDHC_DATA3	DMA_DDONE_B[0]
	MUX_SDHC_WP	SDHC_WP	DMA_DREQ_B[0]
	MUX_SDHC_CD	SDHC_CD	DMA_DACK_B[0]

2.15 TDM interface

The BSC9132 chip has two ports of the TDM interface. Feature supported are:

- Support for 256 channels
- Six wire interface
- 2-bit/4-bit/8-bit/16-bit word size support
- Shared data link mode, RX and TX share sync, clock and full duplex data
- Support for A-law/u-law is supported for 8-bit channels
- Configurable LSB or MSB first

In BSC9132 QDS, TDM signals are connected to the SLIC device for voice data transfer and also terminated in 6-pin 2x3 header for testing purpose. The diagram below shows the TDM connection of the BSC9132 QDS.

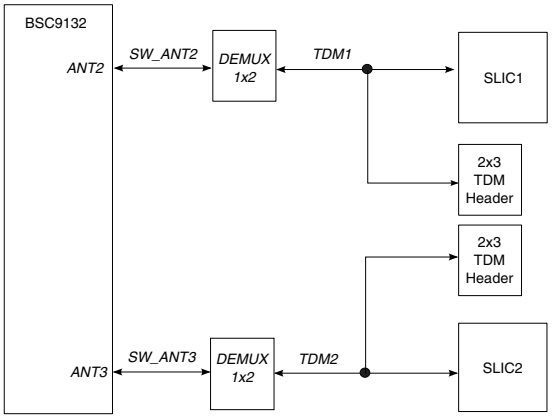


Figure 2-17. BSC9132 QDS TDM connection

Since TDM1 and TDM2 signals are multiplexed with ANT2 and ANT3 interface signals respectively, BSC9132 QDS has an option to select the TDM signals based on the resistor mounting option. The table below shows the TDM signal selection settings.

Table 2-19. BSC9132 QDS TDM1 Interface - assembly option

ANT signals	TDM1 signals	Mux Resisters Install
ANT2_DIO1[0]	TDM1_TCK	R178
ANT2_DIO1[1]	TDM1_TFS	R209
ANT2_DIO1[2]	TDM1_RXD	R191
ANT2_DIO1[3]	TDM1_TXD	R237
ANT2_DIO1[4]	TDM1_RCK	R229
ANT2_DIO1[5]	TDM1_RFS	R233

Table 2-20. BSC9132 QDS TDM2 Interface - assembly option

ANT signals	TDM1 signals	Mux Resisters Install
ANT3_RX_CLK]	TDM2_TCK	R427
ANT3_DIO0[7]	TDM2_TFS	R417
ANT3_DIO0[10]	TDM2_RXD	R415
ANT3_DIO0[11]	TDM2_TXD	R423
ANT3_DIO0[8]	TDM2_RCK	R416
ANT3_DIO0[9]	TDM2_RFS	R425

2.16 Debug support

The debug facilities provided by BSC9132 QDS fall into the following broad categories:

- JTAG-based debugging tools (COP and EONCE JTAG headers)
- SMA-based clock monitoring and injection
- LED-based status monitoring.

2.16.1 JTAG port

The BSC9132 QDS has two JTAG ports. One port is for PA and another port is for DSP. DSP JTAG is available as a multiplexed option.

The primary JTAG port of BSC9132 is PA JTAG. Based on selecting the two CFG_JTAG MODE signals, JTAG is defined. The table below shows the JTAG topology selection.

Table 2-21. BSC9132 JTAG MODE selection

CFG_JTAG_MODE0	CFG_JTAG_MODE0	PA JTAG	DSP JTAG
1	1	Yes	Yes
0	0	Yes	No
1	0	Yes	No
0	1	Yes	NO

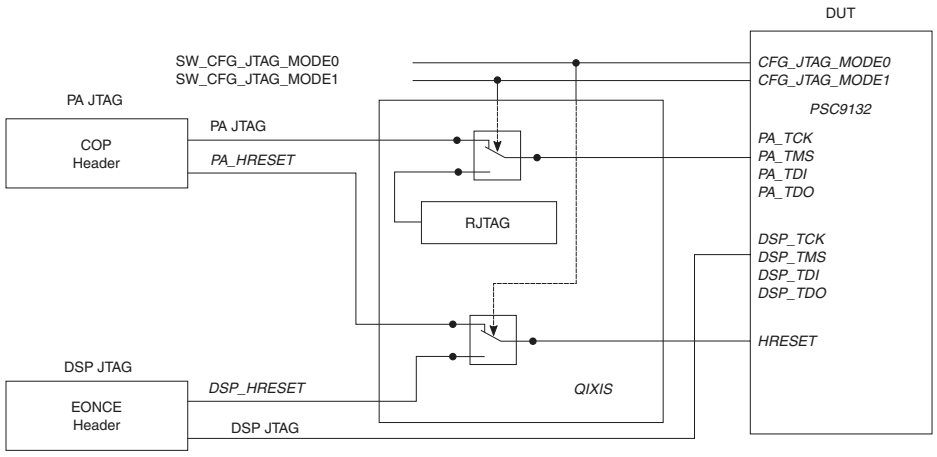


Figure 2-18. JTAG/COP connections

The multiplexer selects from two sources for access to the DUT JTAG port, the legacy COP/JTAG header and EONCE JTAG header. The JTAG reset signal, TRST_B, is managed by QIXIS so that it can be asserted during the HRESET_B interval, so this signal is merged instead of multiplexed.

2.16.2 Clock monitoring and injection

The BSC9132 QDS also provides several SMA (coax) connector locations, which allows monitoring clock signals and in some cases the ability to inject clock signals in place of on-board clock generators.

These connectors are not populated by default and must be installed by the end-user. The type of SMA connector used is Johnson Components Inc. 142-0711-201 SMT.

Once the appropriate connector is installed, Table 2-22 shows the details on converting the board to signal injection, where possible.

Table 2-22. Clock monitoring/injection details

Signal	Description	SMA	SMA Type	Conversion for Injection
DUT_SYSCLK	External Clock source for DUT_SYSCLK buffer.	J31	SMA	Set jumper at the EXT side - J27.[2:3]
DSP_CLKIN	External Clock source for DUT_DSPCLK	J12	SMA	Set jumper at the EXT side - J11.[2:3]
D1_DDRCLK	External Clock source for D1_DDRCLK	J40	SMA	Set jumper at the EXT side - J39.[2:3]
D2_DDRCLK	External Clock source for D2_DDRCLK	J13	SMA	Set jumper at the EXT side - J10.[2:3]
SD2_CLK	External Clock source for SD2_REFCLK	J49	SMA	Remove C148-140. Install R1037,R1039 (0 ohm)
RTCCLK	DUT RTCCLK input	J36	SMA	Set jumper at the EXT side - J30.[2:3]
CP_RCLK	CPRI recovered clock	J48	SMA	n/a - output only

2.16.3 Monitoring LEDs

Lastly, the BSC9132 QDS has numerous LEDs, which can be used to monitor the system status:

Table 2-23. LED Status Monitors

LED	Description
ACTIVE	No HW faults detected, and SW has not asserted a failure.
HRESET	HRESET is asserted - HW or SW HRESET source has asserted.
PASS	No HW faults detected - heart beat blinked.
M[0:7]	During reset and about three seconds thereafter: M[0:3]Power Sequencer State M[4:5]Reset Sequencer State Refer to **QIXIS** HDL source for details. After startup: M0LCS0Boot device activity M1LCS[1:7]Other LB device activity M2I2CMI2C Master (**QIXIS** outbound) activity M3I2CSI2C Slave (**QIXIS** inbound) activity M4(off)(unused) M5DFAILDCM failure (software, unprogrammed) System software can override the LEDs to indicate special status - see Section 5.3.10.
PGOOD	PGOOG is reporting stable of all DUT power.

Table 2-23. LED Status Monitors (continued)

LED	Description
READY	READY is asserted (system is in normal mode) - green
SLEEP	ASLEEP is asserted (system is in deep sleep)

2.17 GPIO controller port

The BSC9132 QDS supports up to 64 GPIO controls that are shared with other important functions. For the BSC9132 QDS, some of the GPIOs are used for other purposes, and so can only be validated with a tester.

The GPIO signals are summarized in Table 2-24.

Table 2-24. GPIO evaluation support summary

GPIO Signal	Shared With	Support	Support Method
GPIO[0:1]	USB_D[4:3]	Yes	Within the FPGA
GPIO[2:3]	USB_D[4:3]	Yes	Accessible at TP64 and TP65
GPIO[5]	ANT4_RX_FRAME	Yes	Accessible at R967 and R968
GPO[6]	ANT4_TX_FRAME	Yes	Accessible at R964 and R965
GPIO[22]	ANT1_DIO1[10]	Limited	Accessible at R299 and R784
GPIO[23]	ANT1_DIO1[11]	Limited	Accessible at R304 and R786
GPIO[42:45]	UART_CTS_B0, UART_RTS_B0, UART_CTS_B1, UART_RTS_B1	Yes	Within the FPGA
GPIO[53]	USB_D[0]	Yes	Within the FPGA, TP62
GPIO[56:57]	UART_SOUT,UART_SIN1	Yes	Within the FPGA
GPIO[62:63]	USB_D[6:5]]	Yes	
GPIO[69:73]	USB_CLK,USB_D7,USB_D2,USB_D1, USB_STP	Yes	
GPIO[80]	ANT1_RX_FRAME	Limited	Accessible at R291 and R775

GPIO[0:7] are the only dedicated GPIO pins, and have several targeted applications and test scenarios. See Section 5.11 for a discussion about the several configurations available. As an overview, refer to read or write GPIO pins (GPIO to GPIO - see Section 3.7, “GPIO module,” on page 3-8)

2.18 DMA controller

The BSC9132 DMA controllers have internal and external controls to initiate and monitor DMA activity. The pins dedicated to DMA functions are multiplexed with other functions and as BSC9132 QDS does not require them for any board-specific devices, the pins are not usually available. For testing purposes, software can configure the DMA signals to become available, and exercise them as described in Table 2-25.

Table 2-25. DMA testing summary

DMA Signals	Shared With	Configuration Required	Test Access
DMA1_DREQ0_B DMA1_DACK0_B DMA1_DDONE0_B	SDHC	SDHC_MUX_SEL_L = 1'b1 (PMUXCR[SHDC_SIM] =01)	Can be controlled and monitored via the **QIXIS** DMA register.
SYS_DMA_REQ SYS_DMA_DONE	ANT4	ANT4_MUX_SEL_L= 1'b1 (PMUXCR2[ANT4] = 01)	Can be controlled and monitored via the **QIXIS** DMA register.

2.19 Temperature monitoring

The BSC9132 has two pins connected to a thermal body diode on the die, allowing direct temperature measurement. These pins are connected to an ADT7461 thermal monitor, which allows direct reading of the temperature of the die and is accurate to ±1 °C.

In addition to software interrupts for thermal measurement, the hardware warning and alarm signals are used to set indicators, and in the case of a thermal alarm, to shut down power to the device. Both the indicators and thermal monitoring is powered by standby power, so thermal shutdown continues even after power is removed, until the thermal monitoring condition is resolved.

The thermal management scheme for the BSC9132 QDS is shown in Figure 2-19.

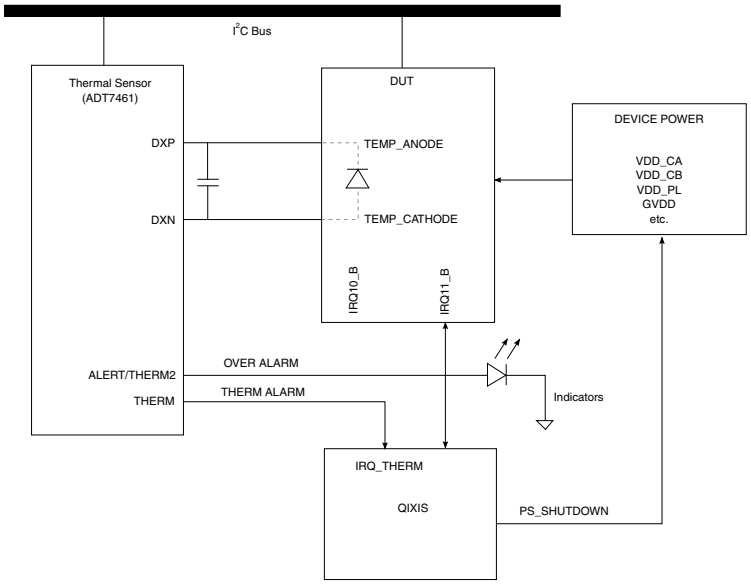


Figure 2-19. Functional block diagram of BSC9132 **QDS thermal management**

Thermal shutdown can be disabled via configuration switch; this is necessary to allow operation of the board with no device installed (making thermal measurements incorrect) or to allow for operation with malfunctioning devices.

2.20 Reset

Reset signals to and from the BSC9132 QDS and other devices on the BSC9132 QDS are managed by **QIXIS**; Figure 2-20 shows an overview of the reset architecture.

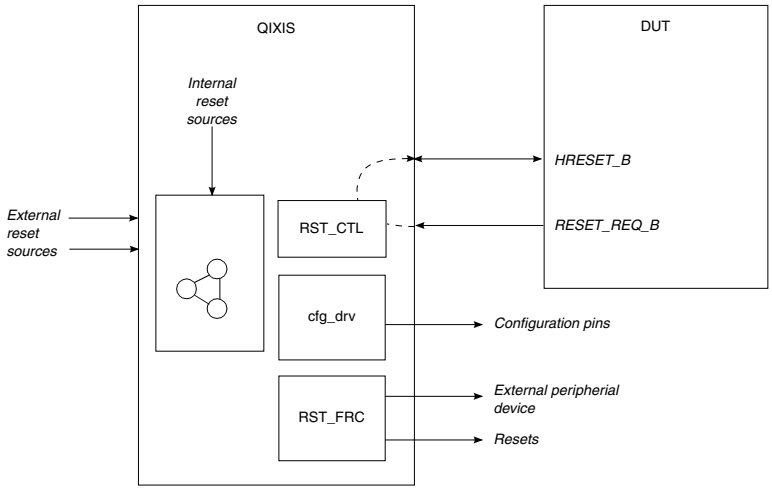


Figure 2-20. Reset architecture

A reset controller (the reset sequencer) manages the collection of various reset triggers, and then asserting reset to internal and external devices, as needed. In addition, the reset controller manages the timing of the pin-sampled configuration driver logic; see Section 5.4.1 for further details.

The reset sequence is triggered by several reasons, including:

- Power up (AUTO_ON mode)
- COP/JTAG tool asserted COP_HRST_B
- HRESET push button
- System or Remote Controller forced a reset via **QIXIS** register write.
- OCM software forced reset
- Reconfiguration watchdog timer expired
- RESET_REQ_B assertion from DUT (determined by RST_CTL[REQMD]; see Section 5.4.2)

While the reset sequencer runs, it resets various internal resources (depending on the reset cause) and signals the OCM and clock programming modules to run. Once complete, the sequencer releases resets asserted to external devices, maintaining configuration-drive signals for the requisite number of SYSCLK periods.

2.21 Clock

The clock architecture of the BSC9132 QDS is driven by the following requirements:

- Easily set SYSCLK, DSPCLK, DDRCLK to popular values.
- Allow SYSCLK to be set to any value from 66-100 MHz.

- Allow DSPCLK to be set to any value from 66-133 MHz.
- Allow D1_DDRCLK to be set to any value from 66-133 MHz.
- Allow D2_DDRCLK to be set to any value from 66-133 MHz.
- Allow SD1_REFCLK to be set to any value from 100-125 MHz.
- Allow SD2_REFCLK to be set to any value from 100-122.88 MHz.
- Allow control of Ethernet clocks by any IEEE-1588 controller module.
- Support synchronous operation of DUT and TDM Interface ports.
- Support synchronous operation of DUT and ULPI Interface ports.
- Support synchronous operation of DUT RTC.

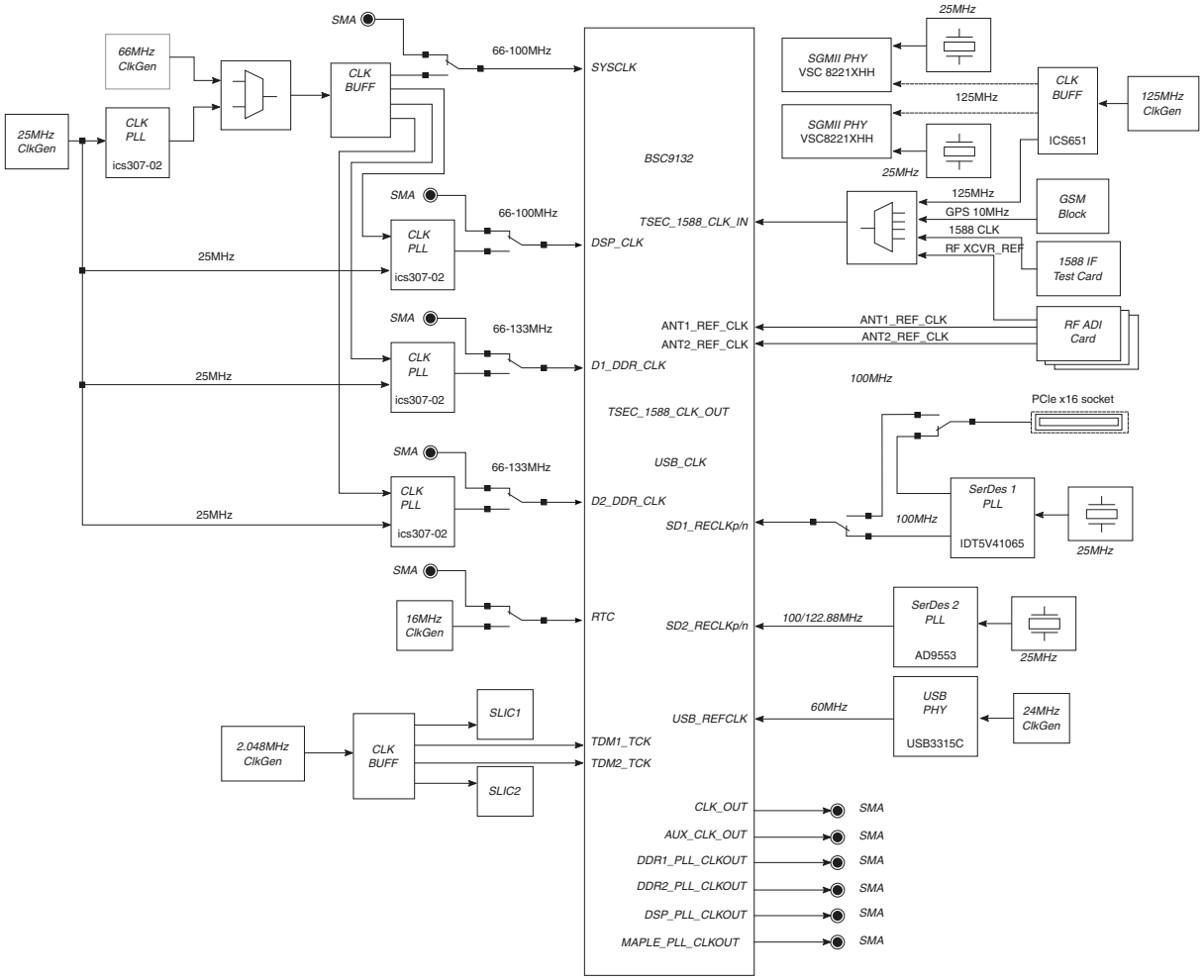


Figure 2-21. Clock architecture

The clocking resources needed for the BSC9132 QDS and for the external peripherals are summarized in the table below.

Table 2-26. BSC9132 QDS clock requirements

Clock	Destination	Clock Frequency	Specs	Type
SYSGEN	BSC9132:SYSCLK SYSCLK Generators REFCLK Generators	25.000	25 ppm	LVTTL
SYSCLK	BSC9132:SYSCLK **QIXIS** SYSCLK HSLB/Testport Card	33–200 MHz	t_R <= 1ns t_F <= 1ns <= 60% duty <= 150 ps jitter	LVTTL
SD1 REFCLK	BSC9132:SD1CLK_DUT(p,n) PEX Slot:SD1CLK_SLOT3(p,n) PHY's: SD1CLK_SGMII(p,n)	100.00 MHz 125.00 MHz	jitter: 80–100 ps skew: 330 ps	LVDS
SD2 REFCLK	BSC9132:SD2CLK_DUT(p,n) Slot 2:SD2CLK_SLOT2(p,n) SD2CLK_SGMII2(p,n)	122.88 MHz	jitter: 80–100 ps skew: 330 ps	LVDS
GTXCLK	BSC9132:EC1_GTX_CLK125 BSC9132:EC2_GTX_CLK125 BSC9132:1588_CLK_IN PHYs:CLK_ENET[1:3]	125 MHz	—	LVTTL
USBCLK	PHYCLK_USBPHY	24.000 MHz	—	LVTTL

2.21.1 SYSCLK

Most of the timing within the BSC9132 QDS is derived from the SYSCLK input. On the BSC9132 QDS, this signal is generated by an IDT ICS307-02 frequency synthesizer, a device which is serially programmed by **QIXIS** as part of the reset/power-up sequence.

The programming value is 24 bits long, and software can set the value to allow any SYSCLK speed within limits; for ease of configuration, **QIXIS** maps 4 switch values into a 24-bit configuration pattern, using the data shown in Table 2-27.

Table 2-27. SYSCLK frequency options

sw_sysclk[0:1]	Nominal SYSCLK	Actual SYSCLK	Error	ICS307 Control Word
0 0 0	66.66 MHz	66.66 MHz	0.0 ppm	0x370801
0 1	100.00 MHz	100.00 MHz	0.0 ppm	0x330801
1 0	133.00 MHz	133.00 MHz	0.0 ppm	0x330a01
1 1	160.00 MHz	160.00 MHz	0.0 ppm	0x310c03
NOTE: The reference clock frequency, 25.000 MHz and control values were carefully chosen to achieve 0.0 ppm error; not all frequencies can achieve this.				

Once generated, SYSCLK is buffered and sent to the following devices:

- the BSC9132 basic clock
- the HSLB/Testport connector high-speed
- **QIXIS** for cycle-accurate reproducibility

2.21.2 DSP clocks

The programming value is 24 bits long, and software can set the value to allow any DSPCLK speed within limits; for ease of configuration, QIXIS maps 4 switch values into a 24-bit configuration pattern, using the data shown in the table below.

Table 2-28. DSPCLK frequency options

sw_dspclk[0:1]	Nominal DSPCLK	Actual DSPCLK	Error	ICS307 Control Word
0 0	66.66 MHz	66.66 MHz	0.0 ppm	0x370801
0 1	100.00 MHz	100.00 MHz	0.0 ppm	0x330801
1 0	133.00 MHz	133.00 MHz	0.0 ppm	0x330a01
1 1	160.00 MHz	160.00 MHz	0.0 ppm	0x310c03
NOTE: The reference clock frequency, 25.000 MHz and control values were carefully chosen to achieve 0.0 ppm error; not all frequencies can achieve this.				

2.21.3 D1_DDR clocks

The programming value is 24 bits long, and software can set the value to allow any DDD1_CLK speed within limits; for ease of configuration, QIXIS maps 4 switch values into a 24-bit configuration pattern, using the data shown in the table below.

Table 2-29. D1_DDR_CLK frequency options

sw_ddr1clk[0:1]	Nominal D1_DDR_CLK	Actual D1_DDR_CLK	Error	ICS307 Control Word
0 0	66.66 MHz	66.66 MHz	0.0 ppm	0x370801
0 1	100.00 MHz	100.00 MHz	0.0 ppm	0x330801
1 0	133.00 MHz	133.00 MHz	0.0 ppm	0x330a01
1 1	160.00 MHz	160.00 MHz	0.0 ppm	0x310c03
NOTE: The reference clock frequency, 25.000 MHz and control values were carefully chosen to achieve 0.0 ppm error; not all frequencies can achieve this.				

2.21.4 D2_DDR clocks

The programming value is 24 bits long, and software can set the value to allow any DDD1_CLK speed within limits; for ease of configuration, QIXIS maps 4 switch values into a 24-bit configuration pattern, using the data shown in the table below.

Table 2-30. D2_DDR_CLK frequency options

sw_ddr2clk[0:1]	Nominal D2_DDR_CLK	Actual D2_DDR_CLK	Error	ICS307 Control Word
0 0	66.66 MHz	66.66 MHz	0.0 ppm	0x370801
0 1	100.00 MHz	100.00 MHz	0.0 ppm	0x330801
1 0	133.00 MHz	133.00 MHz	0.0 ppm	0x330a01
1 1	160.00 MHz	160.00 MHz	0.0 ppm	0x310c03
NOTE: The reference clock frequency, 25.000 MHz and control values were carefully chosen to achieve 0.0 ppm error; not all frequencies can achieve this.				

2.21.5 SerDes clocks

The two SerDes banks each receive an LVDS(HCSL)-compatible differential clock, with the same clock driven to each of the peripherals attached to that same bank. Bank1 - PCI Express and SGMII; bank2 - CPRI.

Each bank's clock source can be independently configured for various clock frequencies, as shown in Table 2-31.

Table 2-31. SerDes SD1 clock frequency options

SW_SD1 [0:1]	SerDes Clock	Protocol Applicability
0 1	100.00 MHz	PCI Express 2.5 Gbps PCI Express 5.0 Gbps SGMII 1.25 Gbps
1 0	125.00 MHz	PCI Express 2.5 Gbps PCI Express 5.0 Gbps SGMII 1.25 Gbps SGMII (3.125 Gbps)
1 0; 11		reserved

Table 2-32. SerDes SD2 clock frequency options

SW_SD2 [0:1]	SerDes Clock	Protocol Applicability
1 0	100.00 MHz	
1 0	125.00 MHz	
00	122.88 MHz	CPRI 6.44Gbps

2.21.6 Ethernet clocks

The basic ethernet clock architecture consists of a precision 125.0 MHz oscillator (+/- 20ppm to allow IEEE-1588-level precision) which is driven to all three Ethernet PHYs as well as the two EC clock inputs on the BSC9132 QDS.

In addition to the above, the ethernet clock system contains a multiplexer which allows an IEEE-1588 controller module to replace the on-board oscillator. When such a card is detected, it is expected to provide the 125.00 MHz reference for the clock buffer. This allows the card to dynamically adjust the clock frequency to maintain time synchronization, which the IEEE 1588 PTP architecture provides.

2.21.7 USB clocks

The four USB PHY devices are clocked from a common 24.0 MHz oscillator and buffer.

NOTE

The USB PHYs can be supplied by 3.3V and 1.8V levels. The buffer drives high levels sufficient to meet the SMC3315 PHY clock input threshold.

2.22 Power

The power supply system of the BSC9132 QDS uses power from a standard ATX PSU to provide the numerous BSC9132, **QIXIS** and peripheral device supplies required. In addition to meeting required power specifications, the following goals guide the power supply architecture:

- DUT_VDD, DUT_VDDC are powered by single supply
- DUT-specific power rails are instrumented such that current measurement is possible
- Automatic collection of voltage, current and power is performed for critical supplies
- All power supplies can be sequenced as per hardware specifications

Note that the BSC9132 QDS generally provides more power than the maximum required to power BSC9132; this allows for the ability to evaluate possible future silicon that may require additional power capacity, and thus should not be used as a guide for customer-specific power supply ratings.

The power supplies provided are organized into the following general categories:

- system power ATX power supply
- DUT core power DUT_VDD, DUT_VDDC
- DUT non-core power BVVD, OVDD, CVDD, G1/2VDD, XPADVDD, SVDD, LVDD, X1/2VDD
- non-DUT power QIXIS hot IO and Core supplies, RF CARDS 1.8V supply

The figure below shows voltage rails connections to meet BSC9132 power supply requirements with ability of the current/voltage monitoring, external voltage source applying to each DUT voltage domain.

Part of the DUT voltage rails are applied to chip through power switches. This allows to control of DUT power sequence flow by using shared voltage regulators between system and DUT.

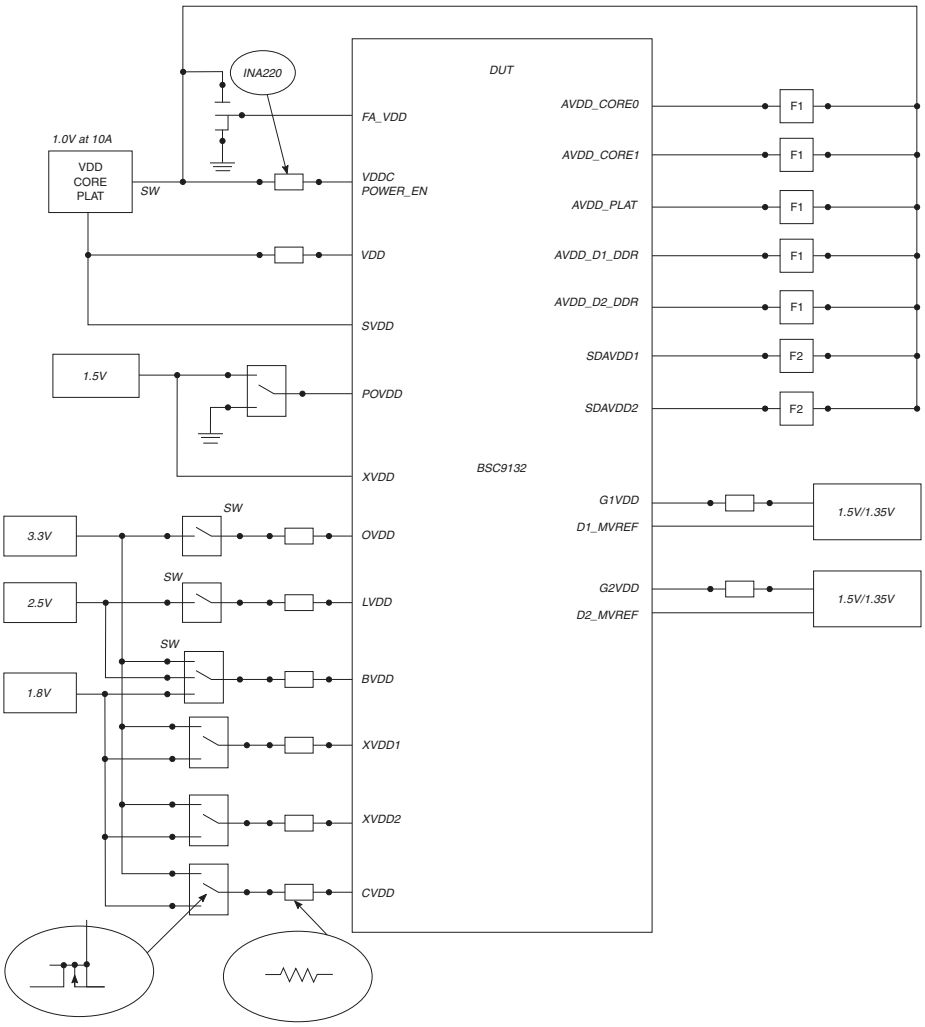


Figure 2-22. BSC9132 QDS DUT power supply block diagram

The diagram below shows the power distribution of the BSC9132 QDS.

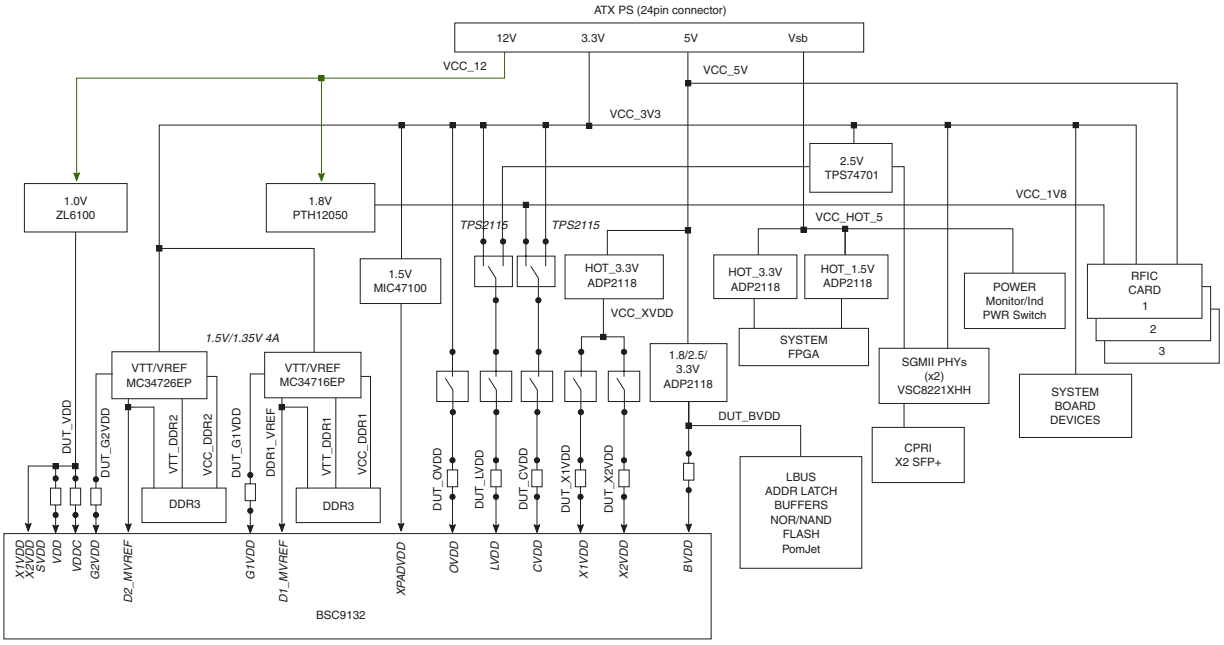


Figure 2-23. BSC9132 QDS power distribution system

2.22.1 System power

The 12V, 5V, and 3.3V power requirements for the BSC9132 QDS are met by the ATX-12V compatible power supply unit (PSU) which is part of the shipped system. The 12V power from the standard ATX header is treated as separate from the ATX-5V power, which supplies a large amount of current and is referred to as “VCC_5V”. The latter is used solely for the VDDC, VDDD power supply rails, while the former is used for miscellaneous purposes, such as fan power and PCI slots.

In addition, the use of a standard ATX PSU affords the ability to use the standby power portion to power **QIXIS**, allowing it to operate fully while all other supplies are off. It can then manage the orderly transition of all other power rails as needed.

Note that to support **QIXIS** standby operation as well as permitting video cards to operate in a PCI Express slot, the PSU should support the following minimum specifications:

- Minimum 450W overall, 500W recommended
- One PCIE 12V connector
- PCIE 12V supports a minimum of 150W
- Minimum 5V, 2A standby current.

WARNING

Be sure to use the ATX power supplied with the system. QorIQ systems do not use as much power from all rails as a typical desktop PC does, and random (unqualified) ATX PSUs are known to not power up without a sufficiently heavy load on all the rails. Grabbing a random PSU from an old computer to power a BSC9132 QDS is definitely not recommended.

2.22.2 Core and platform power

The BSC9132 QDS uses the Zilker Labs ZL6100 switching power controllers. The BSC9132 QDS uses this device as a single-phase controller for up to 22A of power at a nominal 0.9V-to-1.10V output for the core power supplies (DUT_VDD and DUT_VDDC). For each rail, a 0.01 ohm Kelvin resistor and INA220 current monitor is placed in series on core power (DUT_VDDC), allowing accurate current reporting. The BSC9132 QDS also makes use of the PMBus capabilities of the ZL6100 in conjunction with the I2C-based INA220 current monitors to collect Current, Voltage and Temperature (IVT) data on the system in the background (in conjunctions with the QIXIS OCM software).

For each rail, a INA220 current monitor is placed in series with each power, allowing accurate current reporting. The power planes for all three are also provided with plated holes, allowing the following power evaluation options:

- replace on-board supplies, disable on-board supplies, connect bench supplies using plated holes
- precision current measurement by using DVM across plated holes

2.22.3 GVDD/VTT DDR power

The BSC9132 QDS uses the MC34716EP Dual Switch DDR 1.0MHZ 5A synchronous buck regulator, at a nominal 1.5V output. It integrates a synchronous buck controller with embedded power FETs and 1.5A sink/source tracking linear regulator for VTT power and buffered low noise 10mA VREF output. Resistor settings for selecting the DDR3 power supplies are controlled by FPGA through relay. Each of the DDR memory bank includes an own DDR power regulator.

On board where PCB is implemented, separation of the GVDD (chip DDR controller power rail) and VCC_DDRi (DDRi banks power rail), allows to monitor chip DDR controller current voltage value.

2.22.4 Non-DUT power

In addition to the power supplies created for the DUT, there are several that are present to support external peripherals. These supplies are typically enabled but not sequenced in relation to other supplies; Table 2-33 shows a summary of these supplies.

Table 2-33. Miscellaneous Power Supplies

Power Supply	Capacity	Used By
HOT_3.3V	3.3V @ 4A	**QIXIS** IO, PLL
		Standby UART
		I2C EEPROMs
		I2C multiplexers
		PCI Express WAKE# option.
VCC_HOT_1.5V	1.5V @ 4A	**QIXIS** core
VCC_1.8	1.8V @ 6A	DUT: IRS voltage, RF cards[1]
		USB: Core power, HSSI Mux
		SPI: 1.8V devices
VCC_1.2	1.2V @ 1.5A	SGMII PHY Core voltage
VCC_5	5V @ 4A	Voltage monitors, I2C Mux
		SPI: 2.5V devices
VCC_2.5	2.5V @ 1A	SGMII PHY, Clocks
		SPI: 2.5V devices

1 IRS block is a test feature and is not sequenced like other DUT supplies.

Chapter 3
Architecture-Qixis

Figure 3-1 shows a more detailed block diagram of the **QIXIS** FPGA. The principal blocks are described in more detail in the following sections.

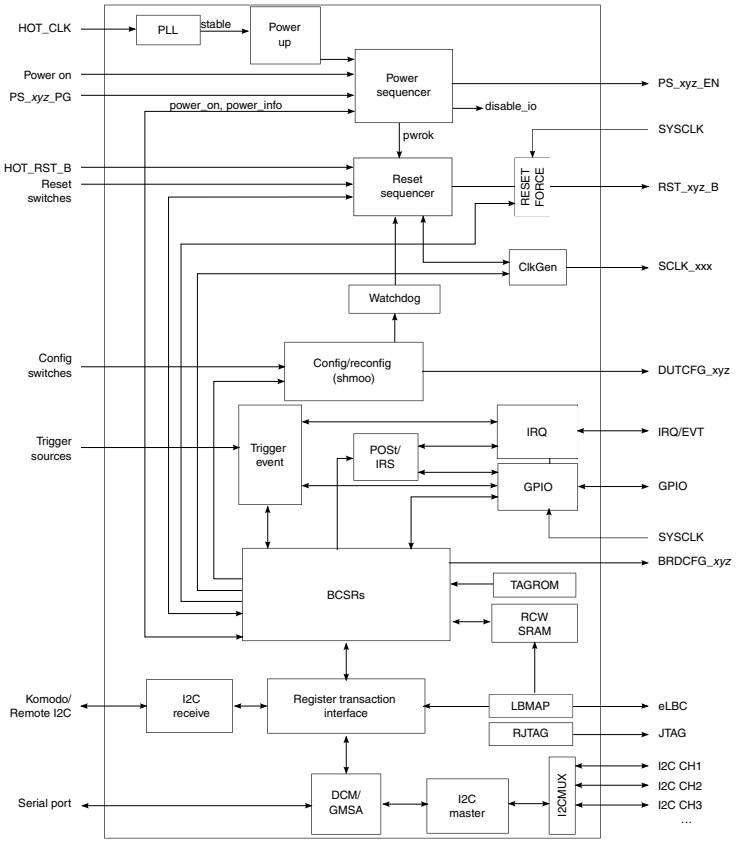


Figure 3-1. QIXIS overview

3.1 BCSR block

The BCSR (Board and Control Register) block is at the center of most of the other blocks of the system. This block supports up to 2048 addresses, though most are not used, and can be sorted into the following broad categories:

- General identification
- FPGA configuration
- Reset control and monitoring

- Power control and monitoring
- Clock control
- Target board config
- Target DUT config
- Reset Control Words (RCW)
- Shared memory
- Remote access assistance

The register file supports concurrent access, as some interfaces (such as the I2C port) do not support hold-off. Although it is expected that only one of the three major interfaces is in use at any one time, the interface is multi-ported and no configuration is needed to enable a port; ports can be switched to as needed.

All registers are reset on power-up, with the exception of SRAM which contains pseudorandom values. Some registers, such as the AUX[1:4] registers, are never modified again; such registers are used by software to preserve values across reset events. Others, such as configuration switch settings may be reset on every HRESET_B/PORESET_B transition. The register definitions describe which reset(s) apply.

Figure 3-2 shows an overview of the register file.

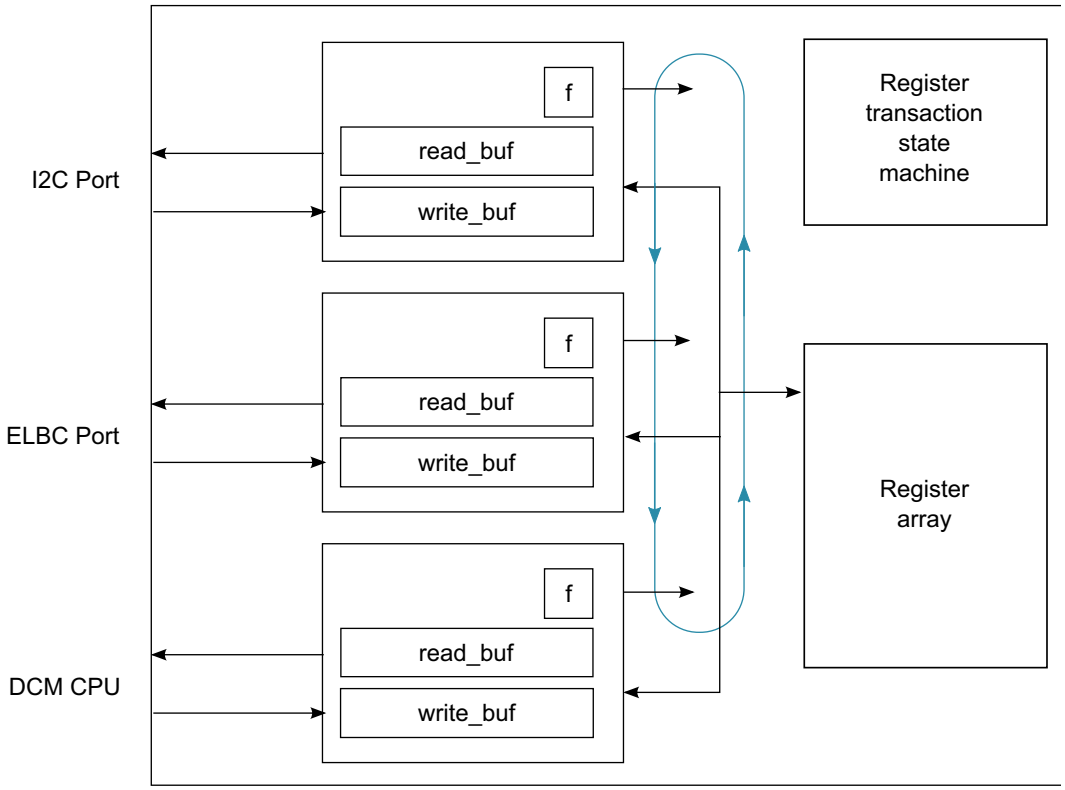


Figure 3-2. Register file overview

The register transaction state machine maintains round-robin access to the register file unit. Writes are cached, with loads bypassing stores as usual (which also prioritizes status monitoring). Not only does this

facilitate shared/concurrent access, it eliminates the blocking that might occur with the relatively slow I2C interface.

3.2 RCW

The Reset Control Word block is a 512 byte SRAM block, whose contents are used to configure the DUT (if the DUT is configured to use it) and rcfg. The RCW is dual-ported to allow access via IFC for the DUT, or from the register file for RCW initialization via I2C or other sources.

To allow the DUT to fetch the RCW from the **QIXIS** SRAM, several initialization operations are needed:

- The DUT must be configured to fetch the RCW from IFC. This is typically the *cfg_rcw_src(0:n)* configuration field of the DUTCFG0 register.
- The SYS_CTL[INTRCW] register must be programmed to force **QIXIS** to intercept IFC accesses to LCS0 during the RCW fetch interval. For most devices, this is the interval from PORESET assertion until ASLEEP is de-asserted.

If the last step is not done, the DUT fetches the RCW from whatever device is mapped to LCS0 on the IFC, typically NOR flash.

All these scenarios assume the DUT is booting via IFC; if it is booting elsewhere (SPI, RAM), **QIXIS** cannot provide the RCW data.

NOTE

The RCW as described here may include a PBL block. This does not affect the RCW operation at all; it is just treated as a particularly large RCW.

3.3 (Re)Configuration

A core requirement of the BSC9132 QDS is to configure both the board and the DUT into a variety of settings, and to later reconfigure the DUT using software alone. The configuration logic collects settings from a variety of sources, uses them to assemble a “working set”, which is then updated and used to configure the target as appropriate. An overview is shown in Figure 3-3.

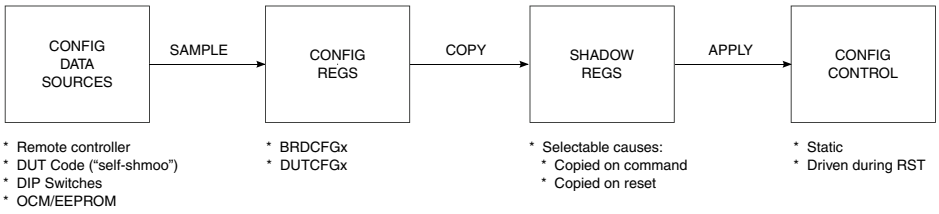


Figure 3-3. QIXIS configuration process

The “shadow registers” perhaps appear new to those familiar with the previous DS configuration logic, but they are not, they are merely formally exposed. Such registers allow for switch registers drive logic to be sampled into switch registers. Such endless loops cause synthesis difficulties, which are solved by the addition of a shadow register.

Variations on this process allow the BSC9132 QDS systems to be easily configured in the two principal configuration environments:

- End-user stand-alone system - used by alpha customers, applications lab.
- Test system remote-controlled system - used by test facilities for fine-grained testing.

The only real difference, in terms of **QIXIS** implementation, is how the configuration values are initially specified.

For the “end-user” system, **QIXIS** sets reasonable defaults, where possible, and uses external DIP switches to control others. In some cases, typical pins such as *cfg_rcw_src[0:n]* are mapped one-to-one; in other cases, such as selecting the SYSCLK frequency, DIP switches are used to map one of 16 pre-defined SYSCLK rates into a 24-bit clock configuration value.

All configuration registers can be changed by software at any time.

3.3.1 Configuration data sources

All configuration registers need a correct value; the default register value is unlikely to always be applicable. Therefore, at system power-up and at varying times thereafter, configuration registers are set and then made available for editing by various controllers. The configuration data sources diagram is shown in Figure 3-5.

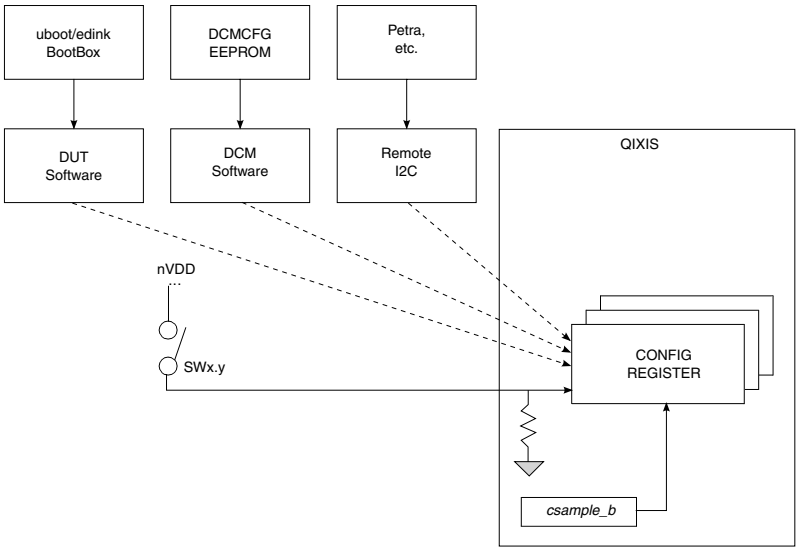


Figure 3-4. Configuration data sources

Configuration is collected in registers called BRDCFGx or DUTCFGx. As part of the reset process, the switches are sampled (except when specifically prevented from doing so) and stored in corresponding registers. After sampling, any of the following may alter the configuration registers:

- I2C remote controller may set/edit values, then allow reset to proceed when ready.
- OCM software copies and/or edits data in an I2C EEPROM into configuration registers.
- DUT software Uboot (rcfg command), BootBox, can all change the configuration.

NOTE

Unlike DS systems, there is not a 1:1 correspondence of DIP switches (SW1:8) and configuration registers (PX_SW[1:8]) - there are too many combinations.

3.3.2 Switch expansion

Switch sampling and mapping is generally one-to-one: that is, if 4 DIP switches are sampled for the RCW location (*cfg_rcw_src[0:3]*), these would typically be latched into 4 bits of a DUTCFG register. In some instances, however, the number of switches becomes unwieldy, so DIP switches are mapped from a few into many bits of a DUTCFG/BRDCFG register.

A particular example of this is the ICS30x serial clock controllers often used for SYSCLK, DDRCLK and others; these require 24 bits of configuration data. Special logic in **QIXIS** maps 4 switches into three 8-bit BRDCFG registers during sampling. The main impact of this is that all configuration logic must deal with the expanded 24-bit data; the 4-bit switch input is not visible at the configuration level. All current DS/BootBox software uses this methodology so no impact is anticipated.

3.3.3 Configuration registers

QIXIS has up to 16 each BRDCFG and DUTCFG registers, allowing up to 128 static and 128 dynamic configuration pins, respectively. Not all of these registers are implemented in every version of **QIXIS**, but the address space is planned to support it.

In addition, fields defined in the BRDCFG/DUTCFG registers are wide enough to accommodate all current QorIQ/PQ/DSP devices. For example, DUTCFG0 allocates 8 bits to `cfg_rcw_loc`, though most current generation devices only require 5. This allows greater software reuse as long as configuration pin changes remain modest.

In terms of behaviour, the actual BRDCFG/DUTCFG registers are simple holding registers, for eventual use with the shadow space.

3.3.4 Shadow registers

The visible configuration registers that are used to configure the target are not directly connected to the configuration pins. Instead, a set of “shadow registers” is actually used to drive the pins, while the visible set of configuration registers is treated as a “working set”, which is applied at a later time. This allows the controller to gradually assemble the configuration data, and eliminates any side effects that might be caused by partial changes.

This step is particularly important for slow target configuration, which may be received over the I2C bus, a relatively slow bus. Coupled with the greater number of configuration registers, converting from USB to ENET mode on a P4080 variant might require 2-3 register changes. By only updating on command, potential conflicts can be avoided.

To achieve this, **QIXIS** always copies data into the shadow register (this is essential for a stand-alone operation) as part of a reset-(re)configuration cycle. It is also possible to initiate a copy at any time, by

setting a **QIXIS** control register. This allows to make changes to the test environment without system power off.

3.3.5 Configuration output

Once the correct configuration values have been set in the shadow registers, **QIXIS** drives the value to the targeted device as required. The way this is done depends on the type of register associated with the configuration pin:

- BRDCFG for static controls; the configuration value selected is always driven.
- DUTCFG for dynamic controls; the value selected is driven during the appropriate interval, then is tri-stated thereafter.

In general, DUTCFG registers are for processor-specific configuration (such as *cfg_rcw_src[0:n]*, and so forth), while BRDCFG registers are for all others (*cfg_spread_spectrum*, perhaps).

The configuration controls are shown in Figure 3-5.

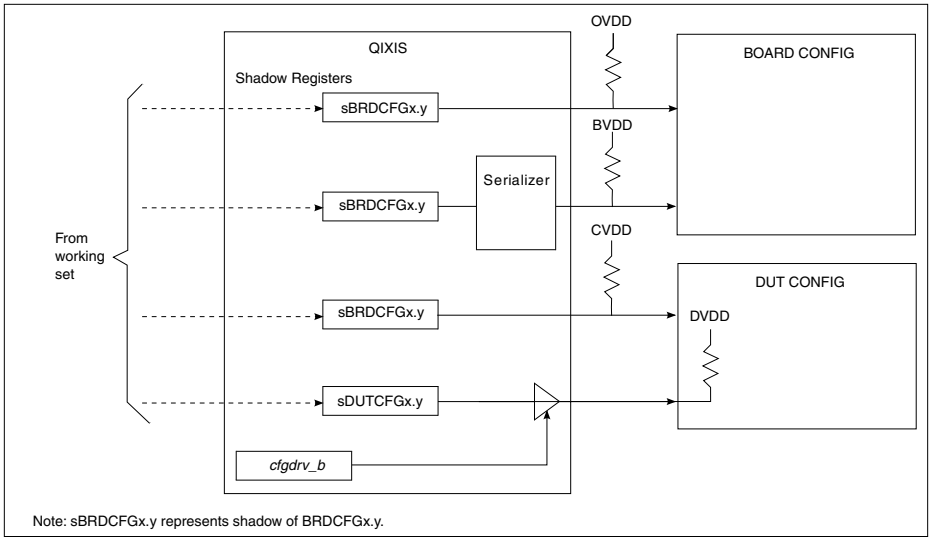


Figure 3-5. Configuration Output Logic

As shown above, the configuration registers are very similar; BRDCFG are always driven, while DUTCFG are typically driven only as needed and are tri-stated at all other times. In some cases, BRDCFG config pins might be used to drive internal FPGA logic, such as the serial clock configuration logic shown above.

NOTE

A few processor pins are static configurations; these are still controlled by DUTCFG registers, but the board-specific implementation of **QIXIS** is expected to handle the details.

Configuration pins need to drive inputs which may reference differing IO levels; the reference IO level may change during any particular test cycle. Therefore, all configuration signals generated by **QIXIS** are open-drain, with no internal pullup. External pullups are expected to set the applicable logic high level (whether explicitly added to the board, or present within the DUT).

3.4 Power control

The power sequencer manages the orderly transition of the various power rails on the system, on command of the user (power switch) or other event (remote system command). The sequencer follows the required ordering specified by each device, and manages transition to/from low-power states, as needed.

QIXIS is responsible for managing power-on/power-off transitions, which can be triggered by the following events:

- Power Switch (on-board or chassis-mount)
- Register bit POWER_CTL[ON] asserted
- AUTO_ON configuration switch high
- OCM processor

Power supplies are grouped into multiple tiers; all power supplies within a tier must report active and stable before the next tier is enabled. A rough example is shown in Figure 3-6.

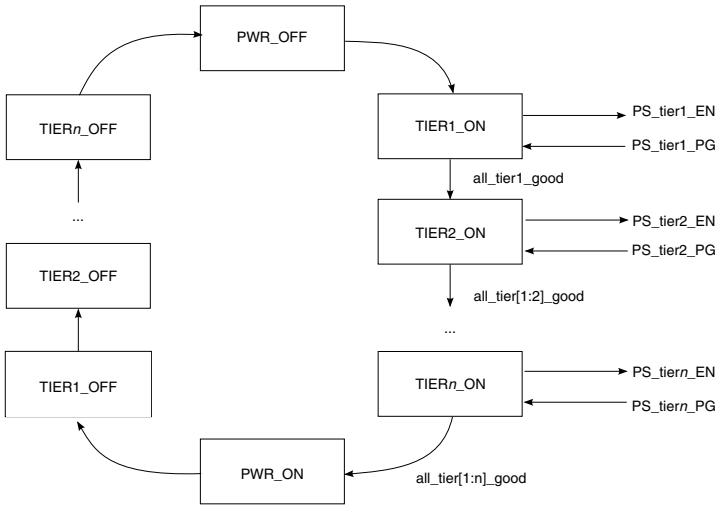


Figure 3-6. Power tiers and sequencing

3.5 Reset

The reset controller manages not only asserting reset to the DUT, but to the rest of the system as well. It also maintains the assertion timing of the configuration drive signal (cfg_drv), which causes the **QIXIS** to drive configuration values onto pin-sampled nets. In operation, it is similar to that of a typical DS with the following additions:

- If the board is configured to “TEST” mode, the reset sequencer halts during reset/configuration assertion, until released via register access (which cannot come via the DUT itself).
- Software can force resets to individual groups of devices at any time.

3.5.1 Reset signal synchronization

In order to preserve cycle-reproducibility, and facilitate test-vector re-use, synchronous timing of various signals relative to SYSCLK is required, and DUT-specific resets fall into that category. Those signals include:

- PORESET
- HRESET

(or equivalent).

However, note that unlike other synchronous signals, reset signals are only *de-asserted* synchronously to SYSCLK. The reason for this is that during initial power up, there is no guarantee that the board has successfully generated a stable SYSCLK, nor that the DUT itself is even ready to use one.

Implementation Note: Resets to the DUT requires special logic to assert without a clock, but deassert using SYSCLK.

3.6 I2C receiver

The I2C receiver block is used to accept transactions from an external I2C-based controller (typically Komodo). The IP maintains a virtual register address which is autoincremented on each successful read or write operation. This address is used to select registers for access, the register block being the only legal target for IO operations.

The IP responds to target address 0x77 (a 7-bit value which does not include the R/W bit). The IP block accepts the following types of transactions:

- address+W, addr, data [, data] write ‘data’ to register at ‘addr’, increment ‘addr’
- address+W, addrsets transfer address to ‘addr’
- address+R, read register at ‘addr’ (previously specified), increment ‘addr’
- address+W, addr, RSTRT, dataread register at ‘addr’, return ‘data’, increment ‘addr’

All other types of transactions (address-only, broadcast) are ignored.

Note that addr is incremented after each read or write operation, allowing streaming of up to 16 sequential bytes in one I2C transaction. Beyond that limit, a separate transaction must be started. An interface gasket manages the connection between the IP block and the register file, loading data only when needed (to avoid pre-caching register values too early).

3.7 GPIO module

The GPIO module manages the GPIO pins, which are used for general purpose testing as well as special-functions (GPIO pins often have numerous additional functions. For this and other reasons, on

BSC9132 QDS platforms, GPIO pins are not used for board control or miscellaneous functions (as might be the case on a reference platform).

For example, on some devices the GPIO[2:3] pins alone can serve as:

- GPIO
- Voltage ID control
- POSt interface
- IRS interface.

In addition to special functions, the triggered event module may also control the outputs.

To manage this complexity, the **QIXIS** internally prioritizes various modules which all share the GPIO signals in a pass-through fashion. Figure 3-7 shows an overview of an example GPIO hierarchy.

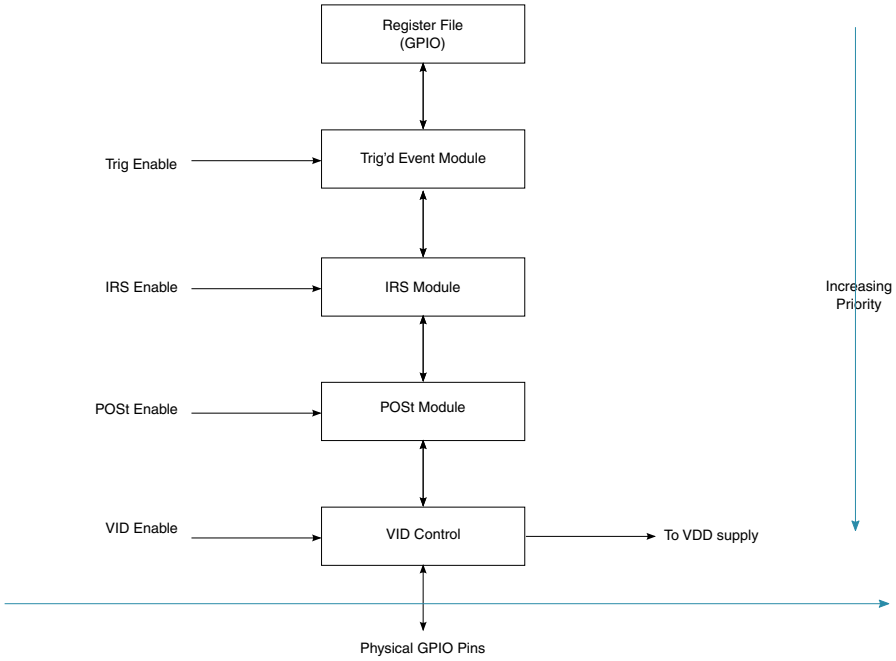


Figure 3-7. GPIO control hierarchy

In the example above, enabling VID controls blocks the use of GPIO for other purposes.Having the POSt module toggle GPIO signals that are also connected to the VDD core supply could have disastrous consequences.

A further complexity is that GPIO multiplexing within the DUT varies widely across even an 8-bit GPIO port, so extensive documentation of the GPIO options is required for each DUT-specific reference manual.

3.8 RemoteJTAG module

The RemoteJTAG module allows remote controllers to query the status of the DUT over the JTAG interface. Note that this interface is very low level - while it might be possible to extract a 65KB LSRL scan chain, it would be extraordinarily slow. VIDControl, which allows access to the VID signals from the

DUT, using them to control the DUT power supplies appropriately. This function is fairly new and standardization is not guaranteed across different DUTs, so consult the specific board documentation for details.

3.9 DCM module

DCM, Development System Control Monitor, is a small 8-bit processor in **QIXIS** that is used to monitor various system parameters, notably Current, Voltage, and Temperature (IVT), while the processor is running applications or test code. By running as a separate process, data collection does not interfere with the DUT, so collected data is not altered by the measurement process.

The DCM module (including the processor) and the application software are not described in this document, but in two separate application notes.

3.10 I2C master

QIXIS has an outbound I2C master block which is used for various purposes:

- DCM processor boot code fetch
- PMBus power supply initialization

Depending on the target system, I2C can connect to one or more I2C domains on the target system, as required for data collection purposes.

Chapter 4
Board Configuration

This chapter describes the necessary steps to configure the board for normal operation.

4.1 Board setup

The board does not contain any headers or other means of setup other than switches. The diagram in Figure 4-1 shows the connections to the board that are needed for bare boards used outside a chassis.

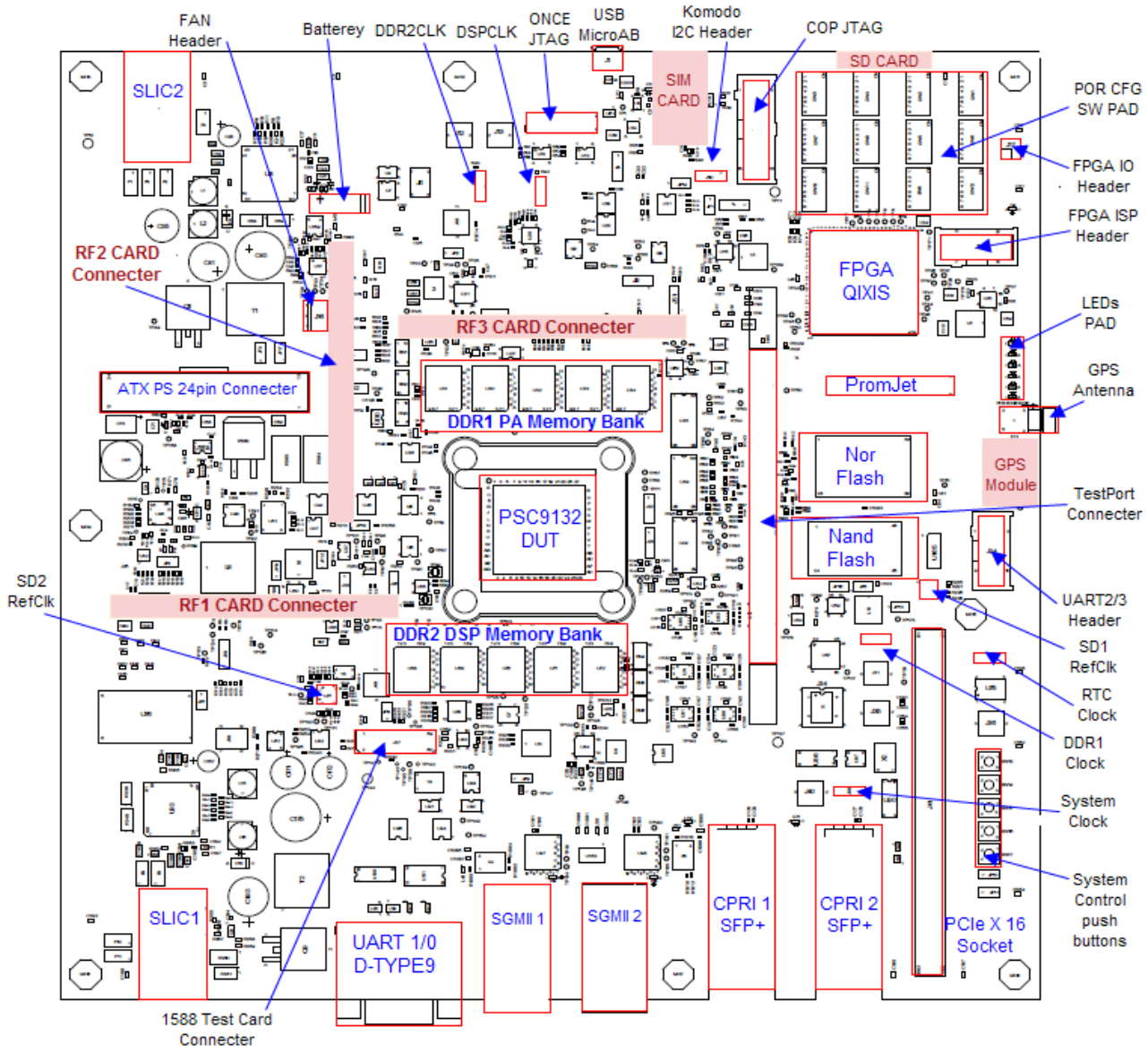


Figure 4-1. Board outline and cabling

4.2 Switch configuration

While the BSC9132 QDS offers a large amount of configuration capability through the **QIXIS** reconfiguration registers, for stand-alone operation, switches are provided to allow easy configuration of many popular options; in particular, those of interest to software developers as opposed to test facilities.

4.2.1 DUT switch configuration

Table 4-1. DUT configuration - SYSCLK PLL (CCB clock frequency)

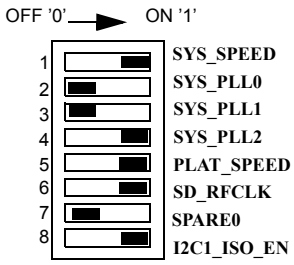
BSC9132 QDS SW3 "SYS_PLL"	SW position notification
	**eSystem Speed** (cfg_sys_speed) 0 -> 33 MHz > SYSCLK frequency < 65 MHz 1 -> 66 MHz > SYSCLK frequency < 133 MHz
	CCB Clock PLL Ratio (cfg_sys_pll[0:2]) 3'b000 - 4:1 3'b001 - 5:1 3'b010 - 6:1 3'b010... - 3'b111 - Reserved
	Platform Speed (cfg_plat_speed) 0 -> 320 MHz > CCB Clock frequency < 167 MHz 1 -> 320 MHz > CCB Clock frequency < 601 MHz
	SerDes Reference Clock Configuration (cfg_srds_refclk) 0 -125 MHz reference clock frequency. 1 -100 MHz reference clock frequency.
	SW_SPARE0 0 - Reserved
	I2C1_ISO_EN 0 - Komodo I2C connected to ISO_I2C1 path 1 - Komodo I2C not connected to ISO_I2C1 path

Table 4-2. DUT configuration - core PLL

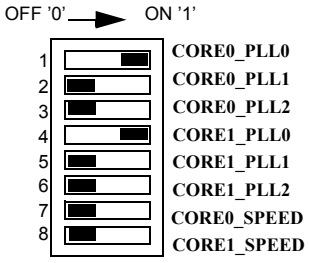
BSC9132 QDS SW4 "CORE_PLL"	SW position notification
	**e500 Core 0: CCB Clock Ratio** (cfg_core0_pll[0:2]) 3'b010 1:1 3'b011 3:2 3'b000 - 4:1 3'b001 - 5:2 3'b100 - 2:1 3'b110 - 3:1 3'b000... 3'b001,3'b111 - Reserved
	e500 Core 1: CCB Clock Ratio (cfg_core1_pll[0:2]) 3'b100 - 2:1
	Core 0 Speed (cfg_core0_speed) 0 -> 500 MHz > Core0 Clock frequency < 1001 MHz 1 -> 1001 MHz > Core0 Clock frequency < 1201 MHz
	Core 1 Speed (cfg_core1_speed) 0 -> 500 MHz > Core0 Clock frequency < 1001 MHz 1 -> 1001 MHz > Core0 Clock frequency < 1201 MHz

Table 4-3. DUT configuration - DDR PLL

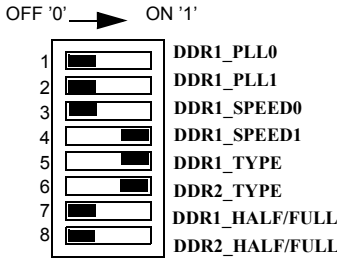
BSC9132 QDS SW1 "DDR_PLL"	SW position notification
	**DDR Complex: DDRCLK Ratio** (cfg_d1_ddr_pll[0:1]) 2'b00 - 8:1 2'b01 - 10:1 2'b10 - 12:1 2'b11 -Reserved
	DDR1 Speed (cfg_d1_ddr_speed[0:1]) DDR1_SPEED0 = 0 PA DDR data rate < 967 MHz DDR1_SPEED0 = 1 PA DDR data rate > 967 MHz DDR1_SPEED1 = 1(see BSC9132RM Table 4-27)
	DDR1 DRAM Type (cfg_d1_dram_type) 0 -DDR3L 1.35V, CKE low at reset 1 -DDR3 1.5V, CKE low at reset
	DDR2 DRAM Type (cfg_d2_dram_type) 0 -DDR3L 1.35V, CKE low at reset 1 -DDR3 1.5V, CKE low at reset
	PA DDR Mode (cfg_d1_ddr_half_full_mode) 0 - full frequency mode -data rate up to 800MHz 1 - half frequency mode -data rate above 800MHz
	DSP DDR Mode (cfg_d2_ddr_half_full_mode) 0 - full frequency mode -data rate up to 800MHz 1 - half frequency mode -data rate above 800MHz

Table 4-4. DUT configuration - DSP PLL

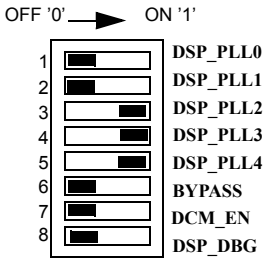
BSC9132 QDS SW2 "DSP_PLL"	SW position notification
	**DSP Subsystem PLL**(cfg_dsp_pll[0:4]) 5'b00111: DDR2_CLK input = 100 MHz (SW11[7:8]) DSP_CLK input = 100 MHz (SW11[3:4]) DSP_Core = 1000 MHz; Maple = 800 MHz DSP DDR Rate = 800Mbps For other available values see BSC9132RM (Table 4-12)
	QIXIS BYPASS Protection (BYPASS) 0 = Disable rotation and over temperature 1 = Normal mode.
	QIXIS DCM Disable 0 - Disable DCM operation 1 = Normal mode.
	DSP DBG Request Mode (DSP_DBG) 0 - Nothing 1 - DSP DBG Request (at the DUT EE0 input)

Table 4-5. DUT configuration - SerDes IO_PORT_SEL

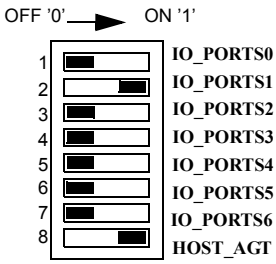
BSC9132 QDS SW5 "SerDes"	SW Position Notification
	**SerDes I/O Port(cfg_srds_io_ports[0:6])** 7'b0010110 - x2 PCIe CPRI2@6.144 CPRI1@6.144 7'b0011011 - x2 PCIe SGMII1@1.25 CPRI1@6.144 7'b0100000 - x2 PCIe SGMII1@1.25 SGMII2@1.25 7'b0100001 - x1PCIe SGMII2@1.25 CPRI2 @6.144 CPRI1@6.144 7'b0100110 - x1PCIe SGMII2@1.25 SGMII1@1.25 CPRI1@6.144 7'b0101011 - SGMII1@1.25 SGMII2@1.25 CPRI2@6.144 CPRI1@6.144 **Host/Agent Configuration cfg_host_agt** 0 - endpoint on PCI Express interface. 1 - root complex on PCI Express interface

Table 4-6. DUT configuration - IFC FCM

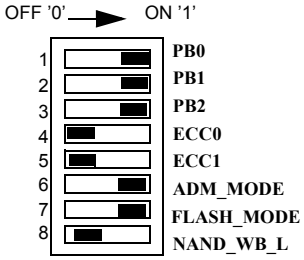
BSC9132 QDS SW7 "IFC"	SW Position Notification
	**IFC Pages Per Block (cfg_ifc_pb[0:2])** 3'b000 - Reserved 3'b001 - 2K pages per block 3'b010 - 1K pages per block 3'b011 - 512K pages per block 3'b100 - 256K pages per block 3'b101 - 128K pages per block 3'b110 - 64K pages per block 3'b111 - 32K pages per block
	IFC ECC (cfg_ifc_ecc[0:1]) 2'b00 ..2'b01 -ECC disabled 2'b10 - 4-bit correction 2'b11 - 8-bit correction
	IFC Address Shift Mode (cfg_ifc_adm_mode) 0 -Lower order address bits are muxed with data on IFC_AD[0:15] 1 -Higher order address bits are muxed with data on IFC_AD[0:15]
	IFC Flash Mode (cfg_ifc_flash_mode) 0 - For NOR, multiplexed NOR flash (AVD type). 1 - For NOR, normal async NOR flash. For NAND, Bad Block Indicator is at page 0 and page 1 of each block.
	Nand Flash Data Bus Width (NAND_WB_L) (Note:for in Rev 2-4 only) 0 - 8-bit Nand Flash Data Bus Width 1 - 16-bit Nand Flash Data Bus Width

Table 4-7. DUT configuration - boot memory location

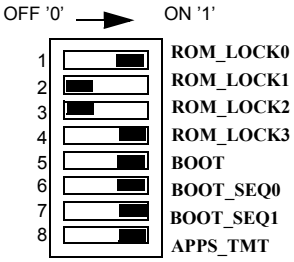
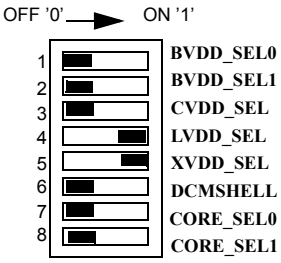
BSC9132 QDS SW8 "BOOT"	SW Position Notification
	**Boot ROM Location** (cfg_rom_loc[0:3]) 4'b0000 - PCI Express 4'b0001..4'b0101 Reserved 4'b0110 - eSPI 4'b0111 - eSDHC 4'b1000 - 8-bit NAND-512b page size 4'b1001 - 8-bit NAND-2K page size 4'b1010 - 8-bit NAND-4K page size 4'b1011 - 8-bit NOR 4'b1100 - 16-bit NAND-512b page size 4'b1101 - 16-bit NAND-2K page size 4'b1110 - 16-bit NAND-4K page size 4'b1111 - 16-bit NOR
	BOOT Enable (cfg_boot_en) 0 - Boot Disable 1 -Boot enable
	Boot Sequencer (cfg_boot_seq[0:1]) 2'b00 - Reserved 2'b01 - Normal I2C addressing mode is used. 2'b10 - Extended I2C addressing mode is used 2'b11 - Boot sequencer is disabled.
	Application/TMT Mode (APPS_TMT) 0 - Test mode (TMT) 1 -Normal Mode (APPS)

Table 4-8. DUT configuration - supply voltage selection

BSC9132 QDS SW6 "VSEL"	SW position notification
	**DUT BVDD Voltage Selection** (BVDD_SEL[0:1]) 2'b00 - 3.3V 2'b01 - 2.5V 2'b10 - 1.8V 2'b11 - Reserved
	DUT CVDD Voltage Selection (CVDD_SEL) 0 - 3.3V 1 - 1.8V
	DUT LVDD Voltage Selection (LVDD_SEL) 0 - 3.3V 1 - 2.5V
	DUT XVDD Voltage Selection (XVDD_SEL) 0 - X1VDD = X2VDD = 3.3V (auto switch to 1.8V when at least one from three RF cards was plugged - Rev3 board) 1 - X1VDD = X2VDD = 1.8V
	QIXIS DCMSHELL Mode 0 = QIXIS DCM does not use COM1 1 = QIXIS DCM uses COM1
	CORE Voltage Margin level (core_sel[0:1]) 2'b00 - 1.0V (default) 2'b01 - 1.2V 2'b10 - 1.1V 2'b11 - Reserved

4.2.2 Board switch configuration

Table 4-9. Board configuration - Serdes reference clocks

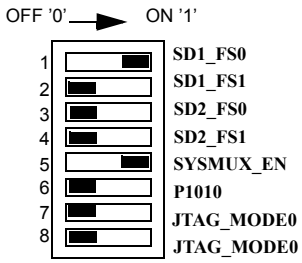
BSC9132 QDS SW10 "SD_CLKSEL"	SW position notification
	**SD1 REFCLK Selection** (sd1_fs[0:1]) 2'b00 - Reserved 2'b01 - 100.00 MHz 2'b10 - 125.00 MHz 2'b11 - Reserved
	SD2 REFCLK Selection (sd2_fs[0:1]) 2'b00 - 122.88 MHz 2'b01 - 125.00 MHz 2'b10 - 100.00 MHz 2'b11 - Reserved
	SYSMUX_EN 0 - SYS_CLK is fixed at 66.67MHz (on board OSC) 1 - P1010/BSC9132 Interposer mode
	P1010/BSC9132 Interposer 0 - BSC9132 Mode 1 - P1010/BSC9132 Interposer mode
	JTAG Topology (cfg_jtag_mode[0:1]) 2'b00 - JTAG_COP => PA_TAP => DSP_TAP 2'b01 - Reserved 2'b10 - JTAG_COP => PA_TAP 2'b11 - JTAG_EONCE => DSP_TAP; JTAG_COP => PA_TAP

Table 4-10. Board configuration - DUT input clocks

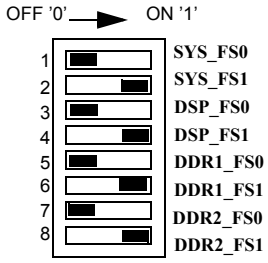
BSC9132 QDS SW11 "SYSCLK_SEL"	SW position notification
	**SYSCLK Frequency Selection** (sys_fs[0:1]) 2'b00 - 66.66 MHz 2'b01 - 100.00 MHz 2'b10 - 133.00 MHz 2'b11 - 160.00 MHz
	DSP_CLK Frequency Selection (dsp_fs[0:1]) 2'b00 - 66.66 MHz 2'b01 - 100.00 MHz 2'b10 - 133.00 MHz 2'b11 - 160.00 MHz
	DDR1_CLK Frequency Selection (ddr1_fs[0:1]) 2'b00 - 66.66 MHz 2'b01 - 100.00 MHz 2'b10 - 133.00 MHz 2'b11 - 160.00 MHz
	DDR2_CLK Frequency Selection (ddr2_fs[0:1]) 2'b00 - 66.66 MHz 2'b01 - 100.00 MHz 2'b10 - 133.00 MHz 2'b11 - 160.00 MHz

Table 4-11. Board configuration - IFC boot remapping

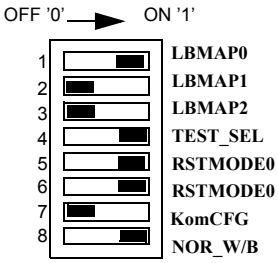
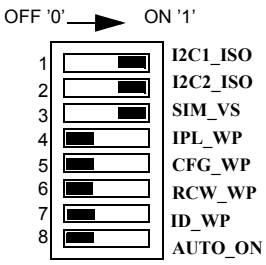
BSC9132 QDS SW9 "DBG"	SW position notification
	**IFC CS0 Mapping** (LBMAP[0:2]) 3'b000 - NOR=CS0;Nand=CS1;QIXIS = CS2 3'b100 - NOR=CS1;Nand=CS0;QIXIS = CS2 3'b111 - PJET=CS0;Nand=CS1;QIXIS=CS2 3'b101 - SRAM_DIMM =CS1; QIXIS=CS2 All other values - Reserved
	TEST Selection (TEST_SEL_B) 0 = with maple 1 = no maple
	QIXIS Reset Request Mode Config (RSTMODE[0:1]) 2'b00 - Do nothing (ignore). 2'b01 - Reserved 2'b10 - Assert HRESET (feedback) 2'b11 - Restart system.
	Komodo to DUT I2C connection sel (KomCFG) 0 - Komodo is connected to DUT I2C1_ISO bus 1 - Komodo is connected to DUT I2C2_ISO bus
	Nor Flash Data Bus Width (NOR_WB_L) (Note: for in Rev 2-4 only) 0 - 8-bit Nor Flash Data Bus Width 1 - 16-bit Nor Flash Data Bus Width

Table 4-12. Board configuration - I2C device write protection

BSC9132 QDS SW12 "WP"	SW position notification
	**I2C1_ISO Enable** (I2C1_ISO) 0 - I2C1_ISO bus is not connected to DUT I2C1 1 - I2C1_ISO bus is connected to DUT I2C1
	I2C2_ISO Enable (I2C1_ISO) 0 - I2C2_ISO bus is not connected to DUT I2C2 1 - I2C2_ISO bus is connected to DUT I2C2
	SIM_VS (SW_CARD for Rev3 board) 0 - Mapping at the J17.6 slave_atx_on output Mapping at the J17.8 slave_hrst_B output On board QIXIS FPGA I2C address is 0x76 1 - Do nothing On board QIXIS FPGA I2C address is 0x66
	QIXIS DCM Program EEPROM Write Protect (IPL_WP) 0 - Write Protect is OFF 1 - Write Protect is ON
	QIXIS CFG EEPROM Write Protect (CFG_WP) 0 - Write Protect is OFF 1 - Write Protect is ON
	DUT BOOT EEPROM Write Protect (RCW_WP) 0 - Write Protect is OFF 1 - Write Protect is ON
	Board ID BOOT EEPROM Write Protect (ID_WP) 0 - Write Protect is OFF 1 - Write Protect is ON
	Board Power ON mode (AUTO_ON) 0 - Power ON from "Power" SW15 push button 1 - Power ON when ATX PS is ON

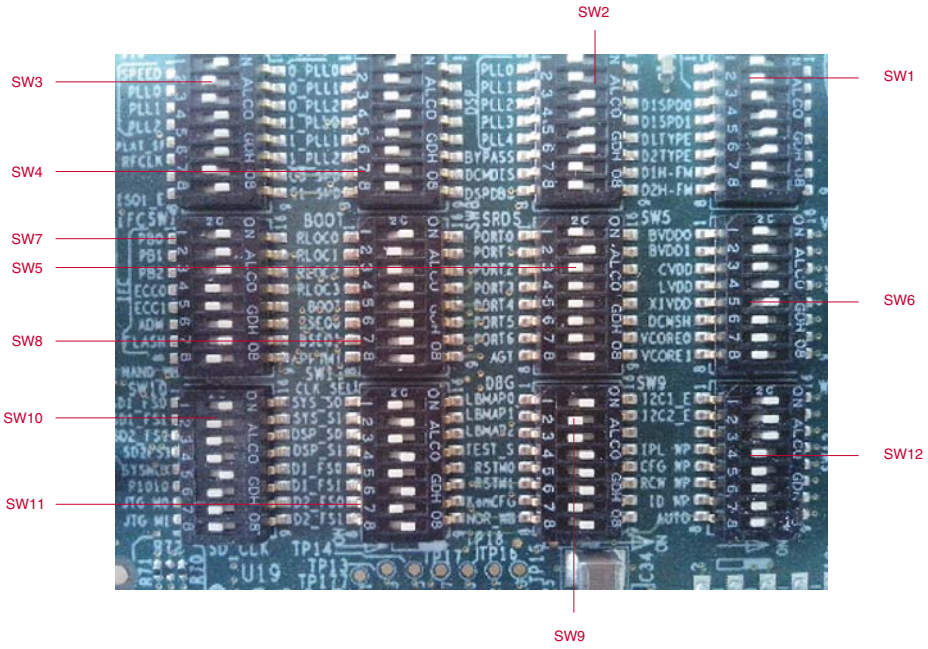


Figure 4-2. BSC9132 QDS DIP-switch locations

4.3 Connector default settings

Table 4-13 lists factory default connector and socket settings for the BSC9132 QDS board. Figure 4-3 notes connector locations.

Table 4-13. Jumpers and connecter default setting

Connector	Name/Function	Type	Features	Description
P1	SIM CARD READER	SIM CARD	SIM CARD	SIM Card Connection
P3	TestPort Connecter	DIMM -144pin;	SMT	Freescale Proprietary
P2	Telephone Conn TDM2(SLIC1)	RJ-11 DUAL RA	2xRJ11-6	T1/E1 Link cable connection
P4	Telephone Conn TDM1(SLIC1)	RJ-11 DUAL RA	2xRJ11-6	T1/E1 Link cable connection
P5	SGMII1	RJ-45;	LED_GETH	Ethernet Link connection
P6	SGMII2	RJ-45;	LED_GETH	Ethernet Link connection
J1	USB2 OTG	Micro-AB USB	9-pin	RF card accommodation
J2	ANT3/4 RF	CON 2X90	SMT	USB OTG cable

Connector	Name/Function	Type	Features	Description
J3	JTAG COP	Header	2x8-pin	External JTAG connection
J5	JTAG EONCE	Header	2x8-pin	External DSP JTAG
J6	TDM2 Clocks options	Header	2x3 pin	**[Default] OPEN**
J7	Komodo I2C bus remote programmer	Header	1x3-pin	External I^2C1 remote programmer connection
J8	Remote I2C1 ISO programmer	Header	1x3-pin	**Not Populated**
J12	External DDR2 clock source	SMA COAX	—	**[Default] OPEN**
J13	External DSP clock source	SMA COAX	—	**[Default] OPEN**
J14	FPGA ISP	Header	2x10-pin	FPGA Program Header
J16	CPU FAN	Header	1x3-pin	12V FAN
J18	ANT2 RF	CON 2X90	SMT	**[Default] OPEN**
J19	PROMJet	Header	2x25-pin	External PROMJet Flash emulator connection
J20	ATX-PS	Connector	2x12-pin	External ATX-PS connection
J24	UART3/4	Header	2x5-pin	**[Default] OPEN**
J26	I^2C1 Zilker converters remote programmer	Header	1x3-pin	External I^2C1 remote programmer connection
J28	I^2C remote programmer	Header	1x3-pin	External I^2C remote programmer connection
J31	External system clock source	SMA COAX	—	**[Default] OPEN**
J36	External RTC clock source	SMA COAX	—	**[Default] OPEN**
J37	1588 riser card	Header	2x30-pin	External 1588 riser card connection
J38	TDM1 Clocks options	Header	2x3 pin	**[Default] OPEN**
J39	External DDR2 clock source	SMA COAX	—	**[Default] OPEN**
J40	External DDR1 clock source	SMA COAX	—	**[Default] OPEN**
J41	PCIe x16 Socket	Socket	164-pin	External PCIe Card
J42	CPRI 1	CON/SFP CAGE	For SFP+ module plugging	Finisar FTL8528P2BCV
J43	CPRI 2	CON/SFP CAGE	For SFP+ module plugging	Finisar FTL8528P2BCV
J44	Dual RS-232: • UART1-Bottom • UART2-Top	DB9 RS-232	Dual 9-pins	External RS-232 cable connection

Connector	Name/Function	Type	Features	Description
J45	SD CARD	CRD SKT SMT		**[Default] No inserted SD/eMMC card**
J49	GPS Antenna Connecter	SMA COAX	—	**[Default] OPEN**
U38	NOR Flash socket for	Socket	56-pin	1GB NOR Flash inserted
U50	NAND Flash socket	Socket	48-pin	4GB NOR Flash inserted

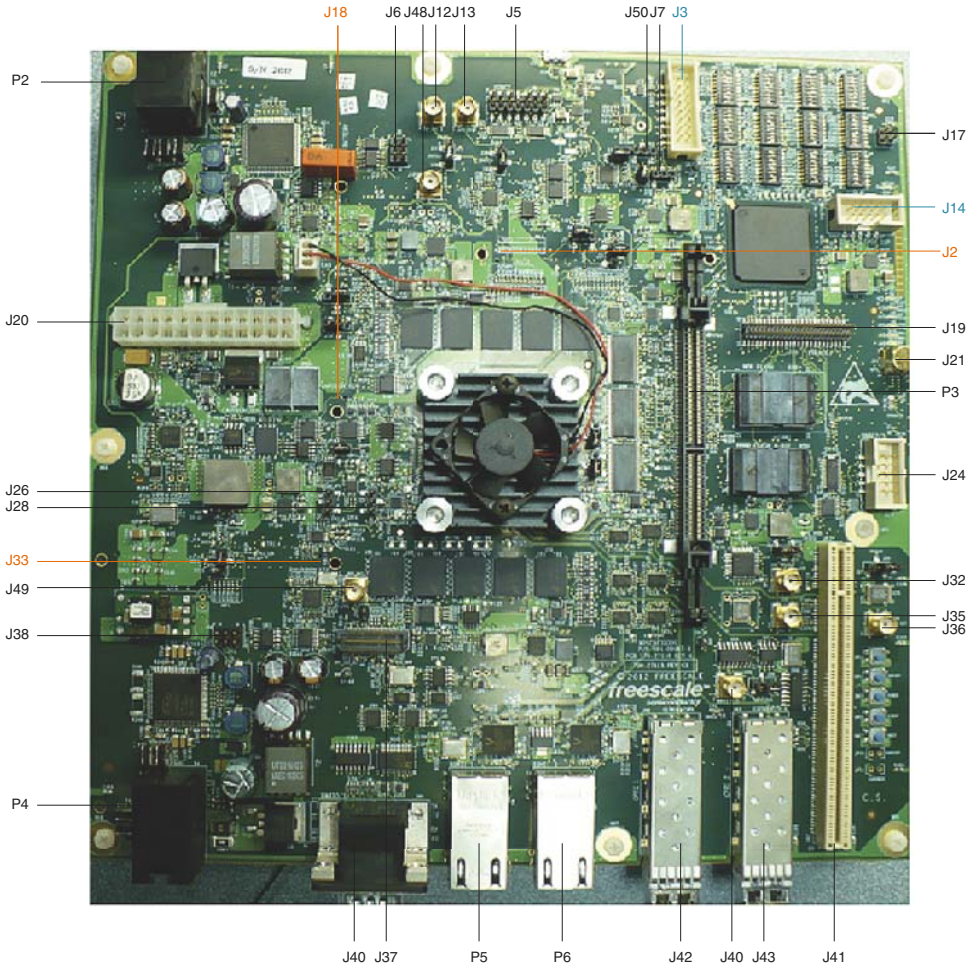


Figure 4-3. BSC9132 QDS connecter locations

4.3.1 Jumper default settings

Table 4-14 lists factory default jumper settings for BSC9132 QDS board. Figure 4-4 notes jumper locations (in blue font)

Table 4-14. Default jumper settings for BSC9132 QDS board

Jumper	Type	Features	Name/Function	Description
JP1	Header	1x2-pin	I2C1_ISO/I2C2_ISO to FPGA connection	• 1-2: [Default] I2C1_ISO connected to through KomodoI2C link to FPGA
J6, J38	Combined Headers	2x3-pin	TDM1,TDM2 SYNC,CLK optional	• [Default] OPEN
J9	FDD/TDD RF3 card mode selection	Header	1x3-pin	• 1-2: TDD mode • 2-3: FDD mode [Default]
J15	FDD/TDD RF2 card mode selection	Header	1x3-pin	• 1-2: TDD mode • 2-3: FDD mode [Default]
J29	FDD/TDD RF1 card mode selection	Header	1x3-pin	• 1-2: TDD mode • 2-3: FDD mode [Default]
J10	Header	1x3-pin	DSP clock source selection	• 2-3: External DSP_CLK • 1-2: [Default] on board DSP_CLK
J11	Header	1x3-pin	DDR1 clock source selection	• 2-3: External DDR1 • 1-2: [Default] on board DDR1
J27	Header	1x3-pin	System clock source selection	• 2-3: External SYSCLK • 1-2: [Default] on board SYSCLK
J30	Header	1x3-pin	RTC clock source selection	• 2-3: External RTC • 1-2: [Default] on board RTC
J39	Header	1x3-pin	DDR2 clock source selection	• 2-3: External DDR2 • 1-2: [Default] on board DDR1
J22	Header	1x3-pin	SecureFuse POVDD selection	• 1-2: Fuse Write • 2-3: Fuse Read [Default]
J23	Header	1x3-pin	Mem Repair Fuse POVDD selection	• 1-2: Fuse Write • 2-3: Fuse Read [Default]
JP7	Header	1x2-pin	1.8V PS OFF	• [Default] OPEN
JP10	Header	1x2-pin	Force Board Reset	• Connected: Force Reset • Disconnected [Default]: Normal operation
JP11	Header	1x2-pin	Force ATX-ON	• Connected: Force ATX-PS ON • Disconnected [Default]: Normal operation
JP14	Header	1x2-pin	SIM_PWR	• Power to SIM_CARD • 1-2 Closed [Default]
JP17	Header	1x2-pin	RF1-RF2 XCVR_IO sharing options	• [Default] OPEN

Jumper	Type	Features	Name/Function	Description
J51	Header	1x3-pin	RF1-RF2 RefClk sharing options	• [Default] OPEN
J52	Header	1x3-pin	VCVR_REF_SELx selection	• ANT1/ANT2 Ref Clock to DUT • 1-2 Closed [Default]

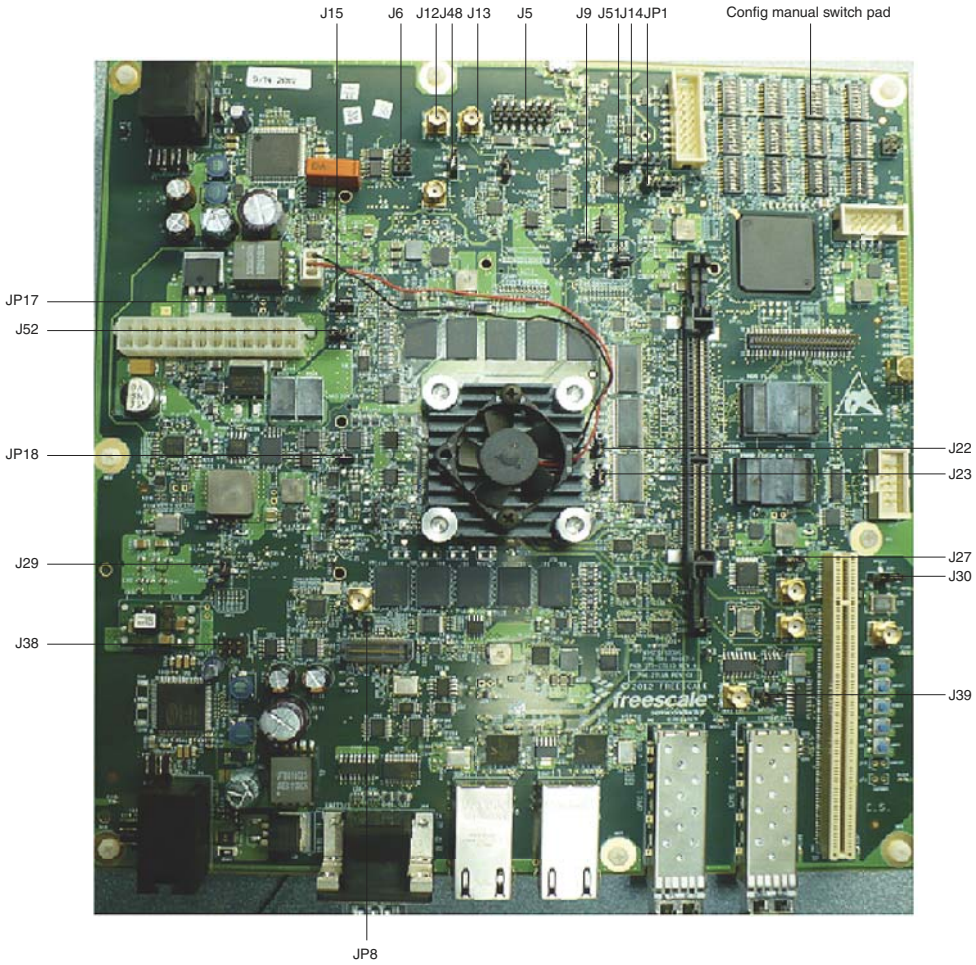


Figure 4-4. BSC9132 QDS jumpers location

4.3.2 Push buttons

Table 4-15 lists the functioning of the BSC9132 QDS board push buttons and switch. Figure 4-2 notes push button and manual (versus DIP-) switch locations.

Table 4-15. BSC9132 QDS push buttons and manual switch

Switch	Push Buttons	Function	Description
SW17	Push Button	DUT HRESET	[Default] ON: Assertion
SW16	Push Button	EVENT	[Default] ON: Assertion
SW14	Push Button	DUT_SRESET	[Default] ON: Assertion
SW13	Push Button	Board reset without power-UP	[Default] ON: Assertion
SW15	Push Button	Board power-UP reset	[Default] ON: Assertion

4.3.3 LEDs lights

Table 4-16. BSC9132 QDS LEDs

LED	Color	Name	LED ON	LED OFF
D3	Red	ORIENT NOT CORRECT	• Incorrect processor orientation in the socket	• Correct processor orientation in the socket
D1	Green	USB_VBUS	• Power Good: USB_VBUS	• Failed: USB_VBUS Power
D4	Yellow	ACTIVE	• POR sequence was over	• POR sequence was not over
D6	Green	DUT_READY	• DUT successful exit from HRESET	• ’
D7	Yellow	DUT_HRESET	• Asserted: BSC9132 HRESET	• BSC9132 HRESET unasserted
D8	Yellow	ASLEEP	• Asserted: BSC9132 ASLEEP	• BSC9132 ASLEEP unasserted
D9	Yellow	SYSTEM RESET	• POR REset asserted	• POR REset unasserted
D10	Yellow	AUX	One of the following: • Successful FPGA initialization; correct processor orientation; and all on-board voltage is in good condition • Register bit PX_CSR[7]=’0’	One of the following: • Power Off • Unsuccessful FPGA initialization • Incorrect processor orientation • Some/all of on-board voltage is in poor condition • Register bit PX_CSR[7]=’1’
D11	Green	DUT PGOOD	• All of DUT Supply voltage rails are OK	• One of DUT Supply voltage rails are not OK
D13	Green	STAT6	• Reset lighted if FPGA OCM is clocked • Normal operation: lighted if register bit PX_LED[6]=’1’	• Register bit PX_LED[6]=’0’

LED	Color	Name	LED ON	LED OFF
D14	Green	SGMCII1 TX_ATIVE	• SGMII1 Downstream is ON	• SGMII1 Downstream is OFF’
D14	Green	SGMII2 TX_ACTIVE	• SGMII2 Downstream is ON	• SGMII2 Downstream is OFF’
LD1	Green	SFP2 TX	• SFP2_TX Good	• SFP2_TX_FAULT
LD2	Green	SFP1 TX	• SFP1_TX Good	• SFP2_TX_FAULT

4.3.4 Clocks and test points

Table 4-17. Clocks and test points

Clock Domain	Schematic Net	Test Point	Expected Value	Remarks
SYS CLOCK	DUT_SYSCLK	TP117	100 MHz	J27 select on board clock source
DDR1 CLK	D1_DDRCLK	J39	100 MHz	J39 select on board clock source
DDR2 CLK	D2_DDRCLK	J.10	100 MHz	J10 select on board clock source
DSP_CLK	DSP_CLKIN	J.11	100 MHz	J11 select on board clock source
RTC_CLK	RTC_SYSCLK	J30	16 MHz	J13 select on board clock source
1588_CLK	1588_SYSCLK	TP139	125 MHz	
CLK_125M_PHY	CLK_125M_PHY[1:2]	U68	125 MHz	
TDM_TCK	TDM_TCK[1:2]	U61.2	2.048 MHz	
USB_CLK_PHY	USB_CLK_PHY	TP57	24 MHz	
DUT_USB_CLK	USB_CLK	U36.30	60 MHZ	
SESDES1 REFCLK	SD1_REFCLK	U53.11:10	100 MHz	
SESDES2 REFCLK	SD2_REFCLK	TP[130:131]	122 MHz	
CON1_XCVR_REF	CON1_XCVR_REF	U37.10	19.2 MHz	
CON1_XCVR_REF	CON2_XCVR_REF	U37.7	19.2 MHz	
ATN1_REF_CLK	ATN1_REF_CLK	U37.12	19.2 MHz	
ATN2_REF_CLK	ATN2_REF_CLK	U37.5	19.2 MHz	

Chapter 5
Programming Model

The **QIXIS** device contains many registers that are accessible from the device over IFC, or remotely over I2C. This chapter explains each of the registers in the Qixis register block. Table 5-1 summarizes the **QIXIS** registers and Table 5-2 shows a detailed address map.

Table 5-1. QIXIS register block map

Base Address Offset	# Registers	Register
0x00	5	Section 5.2, “Identification Registers”
0x05	11	Section 5.3, “Control and Status Registers”
0x10	16	Section 5.4, “Reconfiguration/DCM Registers”
0x20	16	Section 5.5, “Power Control/Status Registers”
0x30	16	Section 5.6, “Clock Control Registers”
0x40	16	Section 5.7, “Reset Control Registers”
0x50	16	Section 5.8, “Board Configuration Registers”
0x60	16	Section 5.9, “DUT Configuration Registers”
0x70	4	Section 5.10, “RCW Access Registers”
0x80	16	Section 5.11, “GPIO Control Registers”
0x9C	4	Section 5.12, “Remote JTAG Access Registers”
0xE0	16	Section 5.13, “Auxiliary Registers”

Table 5-2. QIXIS Register Location Map

Base	+0	+1	+2	+3	+4	+5	+6	+7
0	ID	VER	QVER	MODEL	TAGDATA	CTL_SYS	AUX	CLK_SPD
8	STAT_DUT	STAT_SYS	STAT_ALRM	PRESENT	PRESENT1	RCW_CTL	CTL_LED	I2CBLK
10	RCFG_CFG	RCFG_ST	DCM_AD	DCM_DA	DCMD	DMSG	GDC	GDD
18	DMACK	-	-	-	-	-	-	WATCH
20	PWR_CTL1	PWR_CTL2	-	-	PWR_MSTAT	PWR_STAT1	PWR_STAT2	-
28	-	-	-	-	-	-	-	-
30	CLK_SPD1	CLK_SPD2	-	-	CLK_SYSD0	CLK_SYSD1	CLK_SYSD2	CLK_DSPD0
38	CLK_DSPD1	CLK_DSPD2	CLK_DDR1D0	CLK_DDR1D1	CLK_DDR1D2	CLK_DDR2D0	CLK_DDR2D1	CLK_DDR2D2
40	RST_CTL	RST_STAT	RST_RSN	RST_FRC1	RST_FRC2	-	-	-
48	-	-	-	-	-	-	-	-
50	BRDCFG0	BRDCFG1	BRDCFG2	BRDCFG3	BRDCFG4	BRDCFG5	BRDCFG6	BRDCFG7
58	BRDCFG8	BRDCFG9	-	-	BRDCFG12			

Table 5-2. QIXIS Register Location Map

Base	+0	+1	+2	+3	+4	+5	+6	+7
60	DUTCFG0	DUTCFG1	DUTCFG2	DUTCFG3	DUTCFG4	DUTCFG5	DUTCFG6	DUTCFG7
68	DUTCFG8	DUTCFG9	DUTCFG10	DUTCFG11	DUTCFG12	DUTCFG13		
70	RCW_AD0	RCW_AD1	RCW_DATA	-	-	-	-	-
78	POST_CTL	POST_STAT	POST_DAT0	POST_DAT1	PI_D0	PI_D1	PI_D2	PI_D3
80	GPIO_IO0	GPIO_IO1	GPIO_IO2	GPIO_IO3	GPIO_DIR0	GPIO_DIR1	GPIO_DIR2	GPIO_DIR3
88	-	-	-	-	-	-	-	-
90	-	-	-	-	-	-	-	-
98		-	-	-	RJTAG_CTL	RJTAG_DAT	-	-
A0	TRIG_SRC1	TRIG_SRC2	TRIG_SRC3	TRIG_SRC4	TRIG_DST1	TRIG_DST2	TRIG_DST3	TRIG_DST4
A8	TRIG_STAT	-	-	-	TRIG_CTR0	TRIG_CTR1	TRIG_CTR2	TRIG_CTR3
B0	-	-	-	-	-	-	-	-
B8	-	-	-	-	-	-	-	-
C0	-	-	-	-	-	-	-	-
C8	-	-	-	-	-	-	-	-
D0	-	-	-	-	-	-	-	-
D8	-	-	-	-	-	-	-	-
E0	AUX1	AUX2	AUX3	AUX4	-	-	-	-
E8	-	-	-	-	-	-	AUX_AD	AUX_DA
F0	-	-	-	-	-	-	-	-
F8	-	-	-	-	-	-	-	-

5.1 QIXIS Registers Conventions

An undefined register address does not have any register value. Reads to such addresses should be avoided. If you attempt to read such addresses, undefined data is returned. Writes to such addresses are also undefined. Undefined register addresses may be defined in the future.

For registers with partially-defined bits, reserved bits are read as 0, and should also be written as 0, unless specified otherwise. Any future definition of a reserved bit will endeavour to maintain backward compatibility when the value of 0 is used.

5.2 Identification Registers

This block of registers contain values which identify the board, including major revisions to the board and/or FPGA.

5.2.1 ID Register (ID)

The ID register contains a unique classification number. This ID number is used by EDINK and other software to identify board types. The ID number does not change for any PSC9132QDS revision.

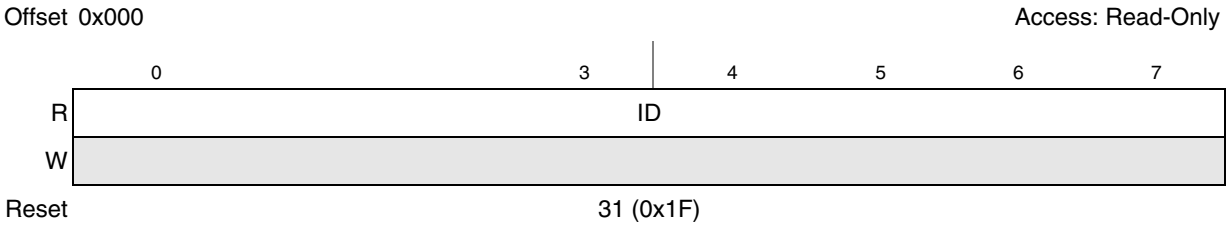


Figure 5-1. ID Register

Table 5-3. ID Register Field Descriptions

Bits	Name	Description
0-7	ID	ID value uniquely identifies each QDS board type. PSC9132QDS: 0x01F (31d)

5.2.2 Version Register (VER)

The VER register records board version information for the PCB board as well the board architecture. The PCB board version can change without impacting the board architecture version, and vice versa, see Table 5-5.

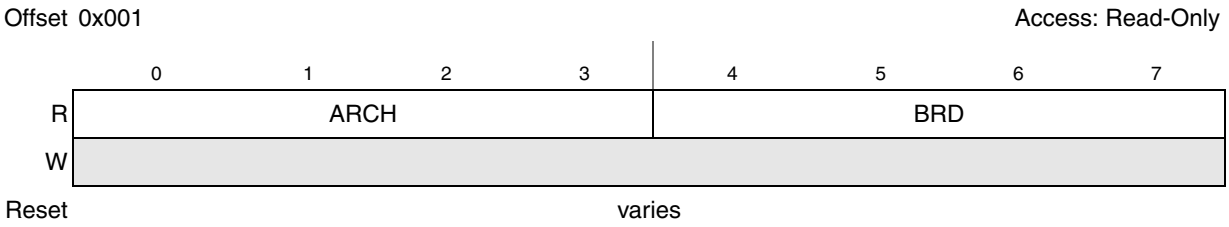


Figure 5-2. VER Register

Table 5-4. VER Register Field Descriptions

Bits	Name	Description
0-3	ARCH	Board architecture Version: 0001 V1 0010 V2 etc.
4-7	VER	PCB board version: 0001 Rev “A” (or pre-release) 0010 Rev “B” 0011 Rev “C” etc.

The fields in this register change only as described in Table 5-5:

Table 5-5. VER Field Change Examples

Change Example	Arch Changes?	BRD Changes?	Commentary
Board changed to move ethernet PHY address by changing resistor options.	Y	N	ARCH changes to inform software there is a change that must be handled. The PCB itself does not change.
PCB artwork changed to increase current capacity of a trace.	N	Y	Arch does not change, this does not affect software.
I2C device relocated from I2C1 to I2C2	Y	Y	This is both a physical and software-impacting change.
Flash device changed from MFG “X” to MFG “Y”	N	N	Software can detect this without assistance.

The ARCH field is used by QIXIS and software to handle architecture changes. The ARCH field allows the use of a common QIXIS image across multiple board revisions, if supported by the device.

The BRD field lets end users determine the version of the board. Software can use this field to print board version identification. For example:

```
printf(" Board Version: %c\n", (get_pixis( VER ) & 0x0F) + 'A' - 1 );
```

NOTE

For details about assembly revision and board errata, refer to the SystemID specification.

5.2.3 QIXIS Version Register (QVER)

The QVER register contains the major version information of the **QIXIS** system controller FPGA.

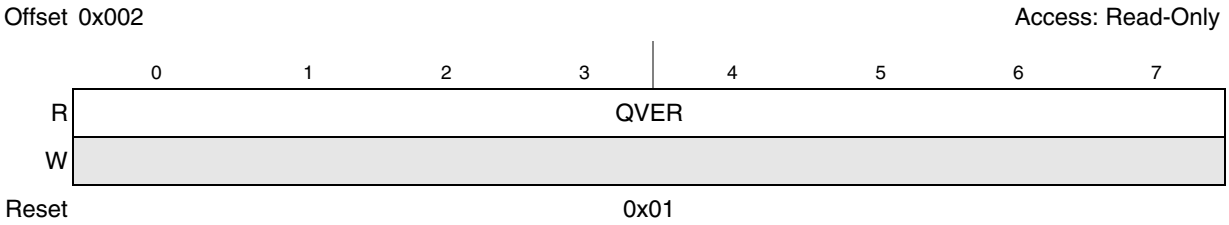


Figure 5-3. QVER Register

Table 5-6. QVER Register Field Descriptions

Bits	Name	Description
0-7	QVER	Incrementing field noting the current **QIXIS**. 0x01 = V1 0x02 = V2 etc.

For information about Qixis minor revision, build date, refer to the TAGROM data.

5.2.4 Programming Model Register (MODEL)

The MODEL register contains information relating to the software programming model version. The MODEL field is updated, if a register or register field is added or updated.

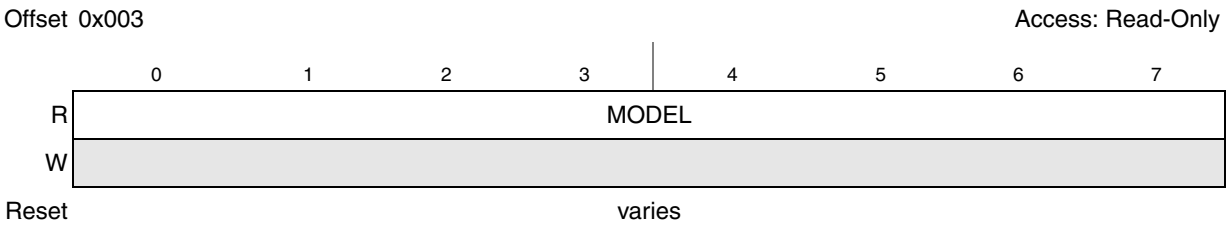


Figure 5-4. MODEL Register

Table 5-7. MODEL Register Field Descriptions

Bits	Name	Description
0-7	MODEL	Incrementing field representing programming model versions: 0x01: V1 0x02: V2 etc.

NOTE

If the FPGA revision changes, the MODEL field is not impacted, for example for internal HW-related changes.

5.2.5 QIXIS Tag Access Register (QTAG)

The QTAG register is used to access the internal TAGROM in each **QIXIS**. This 128-byte ROM contains additional information related to the **QIXIS** image. To access the TAGROM, an address value is written to QTAG, then QTAG is read to obtain the data.

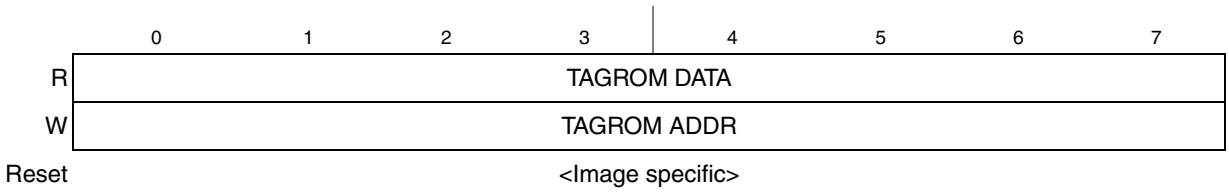


Figure 5-5. QTAG Register

Table 5-8. QTAG Register Field Descriptions

Bits	Name	Description
0-7	ADDR	Read: Data to read from TAG ROM. Write: Address of tag data to read

The TAGROM data can be read using the following code lines:

```
for i = 0 to 127 begin
```

```
qixis_write( PX_TAG, i )
tagrom[i] = qixis_read( PX_TAG )
end
```

The data contained in the TAGROM is organized as follows:

Table 5-9. TAGROM Data Contents

Bytes	Name	Description
0-3	TAG	Validation tag: Must be the value “PXTG” else the remainder is not valid.
4	—	0x00
5	VER	Minor version of FPGA. This value is automatically incremented on each build, and can be used to track engineering builds.
6-7	—	0xFF
8-11	DATE	32-bit build date and time, represented in Unix UTC time.
12-15	—	0xFF
16-63	IMAGE	Null-terminated C-string representing the FPGA image file, such as `QIXIS_PSC9132_0611_1733`.
64-127	—	0xFF

5.3 Control and Status Registers

This block of registers control the operation of **QIXIS** itself (or other operations which do not constitute controlling the board or the DUT, which are managed with BRDCFG/DUTCFG registers) or monitor the status of various things.

5.3.1 System Control Register (CTL_SYS)

The CTL_SYS register is used to control various aspects of the target system.

Offset 0x005 Access: Read-Write

	0	1	2	3	4	5	6	7
R W	TPEN	—			EVTSW		LED	FAIL
Reset	All zeroes							

Figure 5-6. CTL_SYS Register

Table 5-10. CTL_SYS Register Field Descriptions

Bits	Name	Description
0	TPEN	Test-port Enable: 0No testport card is installed; or if so, test-port is specifically disabled. 1A test-port card must be installed in the system, and the DUT should be configured for testport mode. Controls QIXIS TESTPORT_EN output signal.
1-3	—	Reserved.

Table 5-10. CTL_SYS Register Field Descriptions (continued)

Bits	Name	Description
4-5	EVTSW	Event Switch Mapping: 00 EVENT_SW restarts the DCM processor. 01 EVENT_SW asserts IRQ8. 10 Reserved 11 Reserved
6	LED	Software Diagnostic LED Enable: 0 Diagnostic LEDs M0-M7 operate normally. 1 Software can directly control the M0-M7 monitoring LEDs using the PX_LED register value.
7	FAIL	Software Failure Diagnostic LED: 0 FAIL LED is not asserted due to software (it might be on due to hardware failures). 1 FAIL LED is forced on. This indicates a software-diagnosed error.

5.3.2 Auxiliary Register (AUX)

The AUX register is used to store system information. The AUX register is initialized to zero when the system is powered-up. Once initialized, the register value is not changed by QIXIS again.

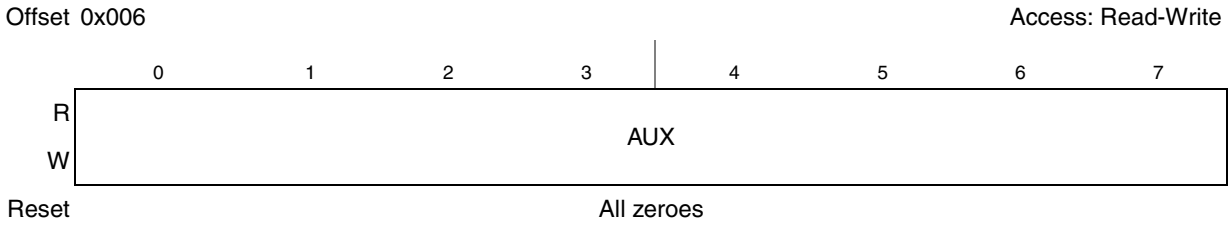


Figure 5-7. AUX Register

Table 5-11. AUX Register Field Descriptions

Bits	Name	Description
0-7	AUX	User-supplied value.

5.3.3 Speed Register (CLK_SPD)

The SPD is a legacy-compatible register used to communicate the current switch selections for SYSCLK and DDRCLK clock settings, if required. These values are used by boot software to specify one of the 8 startup values for each clock and to accurately initialize SYSCLK-dependant parameters, such as those for local bus, DDR memory, and I2C clock rates.

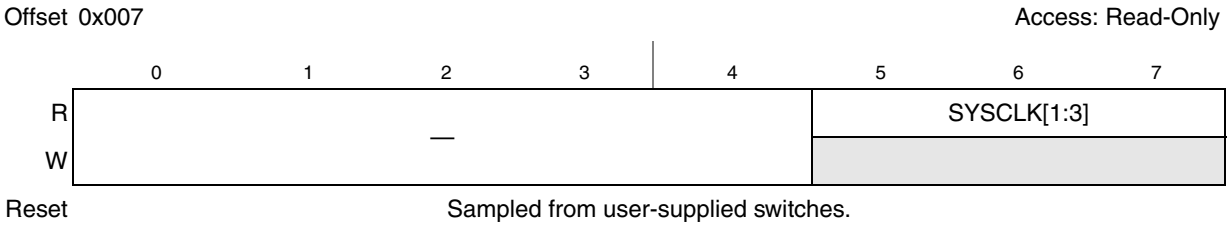


Figure 5-8. SPD Register

Table 5-12. SPD Register Field Descriptions

Bits	Name	Description
0-4	—	Reserved.
5-7	SYSCLK	Reflects switch settings used to preset SYSCLK.

5.3.4 DUT Status Register (STAT_DUT)

The STAT_DUT register is provided for the DUT to communicate status to remote systems. In practice, it is similar to AUX registers, which are cleared on power-up and then preserved thereafter, except:

- Access to this register cannot be blocked from any access path.
- Only DUT can write to this register through IFC bus. Any other FPGA interface (I2C1 and I2C2) can only read it.

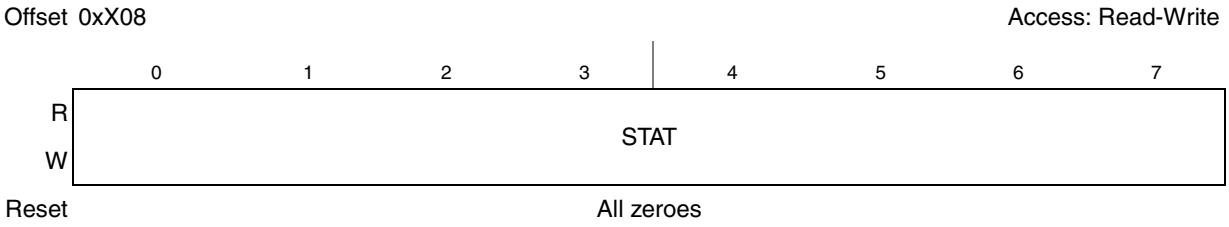


Figure 5-9. STAT_DUT Register

Table 5-13. STAT_DUT Register Field Descriptions

Bits	Name	Description
0-7	STAT	Test Software Defined value.

Because access to this register cannot be blocked, it is present in each 256B block accessed through I2C. See Section 5.3.11, “I2C Block Register (I2C_BLK)” for details.

5.3.5 System Status Register (STAT_SYS)

The STAT_SYS register reports general system status, as described in Table 5-14.

Offset 0x009 Access: Read-Only

	0	1	2	3	4	5	6	7
R	READY	—				RCFG	ASLEEP	TEST
W								
Reset	Zeroes; affected by configuration switches							

Figure 5-10. STAT_SYS Register

Table 5-14. STAT_SYS Register Field Descriptions

Bits	Name	Description
0	READY	DUT Ready Active: 0 Not Ready. 1 DUT has completed its startup sequence.
1-4	—	Reserved.
5	RCFG	Reconfiguration Active: 0 The system has been configured as normal. 1 The system has been reconfigured by software.
6	ASLEEP	ASLEEP Reporting: 0 At least one core is actively operating. 1 All cores are in sleep mode.
7	TEST	TEST/NORMAL Mode: 0 The system is in normal mode. Switches configure the system, and reset operates normally. 1 The system is in test mode. On power-up, the reset sequencer halts to allow remote download of the RCW, configuration, and other parameters. Reset does not proceed until configured to do so.

5.3.6 Alarm Status Register (STAT_ALARM)

The STAT_ALARM register detects and reports any alarms raised in the QIXIS system.

Offset 0x00A Access: Read-Only

	0	1	2	3	4	5	6	7
R	GEN	SGMII_INT1	SGMII_INT2	SLIC1_INT	SLIC2_INT	RTC_INT	1588_INT	TALERT
W								
Reset	Zeroes							

Figure 5-11. STAT_ALARM Register

Table 5-15. STAT_ALARM Register Field Descriptions

Bits	Name	Description
0	GEN	General Alarm Status: 0 No unspecified fault detected. 1 Fault for which no specific additional data is available.
1	SGMII_INT1	SGMII 1 PHY interrupt 0 Not action. 1 Interrupt request Sampled SGMII_PHY1_INT_B input.
2	SGMII_INT2	SGMII 2 PHY interrupt 0 Not action. 1 Interrupt request Sampled SGMII_PHY2_INT_B input.
3	SLIC1_INT	SLIC1 PHY interrupt 0 Not action. 1 Interrupt request Sampled SLIC1_INT_B input.
4	SLIC2_INT	SLIC2 PHY interrupt 0 Not action. 1 Interrupt request Sampled SLIC2_INT_B input.
5	RTC_INT	Real Time controller interrupt 0 Not action. 1 Interrupt request Sampled RTC_INT_L input.
6	1588_INT	1588 Controller interrupt 0 Not action. 1 Interrupt request Sampled AD7998_FPGA_INT input.
7	TALERT	Temperature Alert: 0 The temperature is within normal limits. 1 The temperature has exceeded warning limits. **Note:** This signal corresponds to the ADT7461, and the temperature limits and behaviour (latching vs. transient) depend upon software programming.

5.3.7 Presence Status Register (STAT_PRESENT)

The STAT_PRESENT register reports on the presence of various removable devices.

Offset 0x00B Access: Read-Only

	0	1	2	3	4	5	6	7
R	DUT	P1010	SFP1	SFP2	RF1	RF2	RF3	SLOT1
W								

Reset Depends on hardware installed.

Figure 5-12. STAT_PRESENT Register

Table 5-16. STAT_PRESENT Register Field Descriptions

Bits	Name	Description
0	DUT	PSC9132 processor in socket: 1 A processor is detected in the socket. (always 1) 0 No device detected.
1	P1010	P1010 interposer installed: 0 No interposer detected. 1 A interposer is detected in the slot.
2	SFP1	SFP1 optical transceiver resent: 0 A module is detected in the cage. 1 No module detected.
3	SFP2	SFP2 optical transceiver resent: 0 A module is detected in the cage. 1 No module detected.
4	RF1	ADI RF1 Card: 0 A card is detected. 1 No card detected.
5	RF2	ADI RF2 Card: 0 A card is detected. 1 No card detected.
6	RF3	ADI RF3 Card: 0 A card is detected. 1 No card detected.
7	SLOT1	PCI Express Slot 1: 0 A card is detected. 1 No card detected.

5.3.8 Presence Status Register (STAT_PRESENT1)

The STAT_PRESENT1 register reports on the presence of various removable specify test devices.

Offset 0x00C — Access: Read-Only

	0	1	2	3	4	5	6	7
R	TEST	IEEE	SIM	SD	—			
W								
Reset	Depends on hardware installed.							

Figure 5-13. STAT_PRESENT1 Register

Table 5-17. STAT_PRESENT1 Register Field Descriptions

Bits	Name	Description
0	TEST	TEST Port Card installed: 1 No card detected. 0 A Test Port Card is detected in the slot.
1	IEEE	IEEE-1588 Riser Card installed: 0 A IEEE-1588 controller is detected in the slot. 1 No card detected.
2	SIM	SIM Card installed: 0 A SIM Card is detected in the slot. 1 No card detected.
3	SD	SD Card installed: 0 A SD Card is detected in the slot. 1 No card detected.
4-7	—	Reserved.

5.3.9 RCW Control Register (CTL_RCW)

The CTL_RCW register is used to manage various aspects of the Reset Control Word (including PBL) resources.

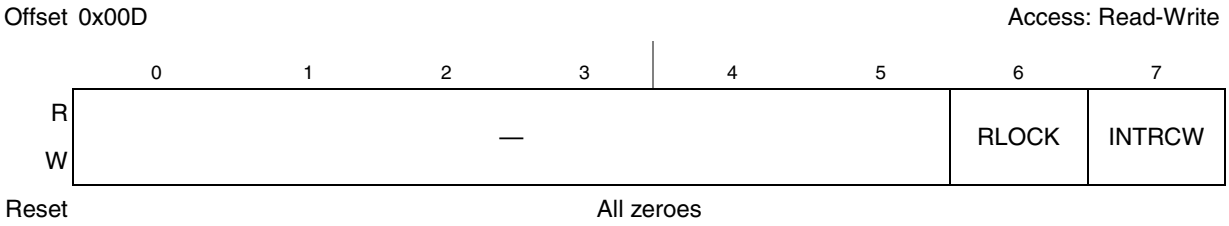


Figure 5-14. CTL_RCW Register

Table 5-18. CTL_RCW Register Field Descriptions

Bits	Name	Description
0-5	—	Reserved.
2	RLOCK	RCW Lock: 0 RCW SRAM can be modified through any path. 1 RCW SRAM cannot be modified.
3	INTRCW	Enable internal RCW SRAM: 0 RCW is expected to be provided by boot device. The internal RCW SRAM is just a 256B SRAM. 1 The internal is expected to contain RCW(+PBL) data. During the assertion of RESET to the DUT, as long as it asserts ASLEEP, accesses to LCS0_B are assumed to be fetches of the RCW word from the IFC. **QIXIS** will redirect LB_LCS0_B to the internal RCW SRAM during this interval, reverting to normal operation thereafter.

5.3.10 LED Register (LED)

The LED register can be used to directly control the monitoring LEDs (M0-M7) for software debugging and messaging purposes. Direct control of the LEDs is possible only when PX_CTL[LED] is set to 1; otherwise **QIXIS** uses them to display other activity.

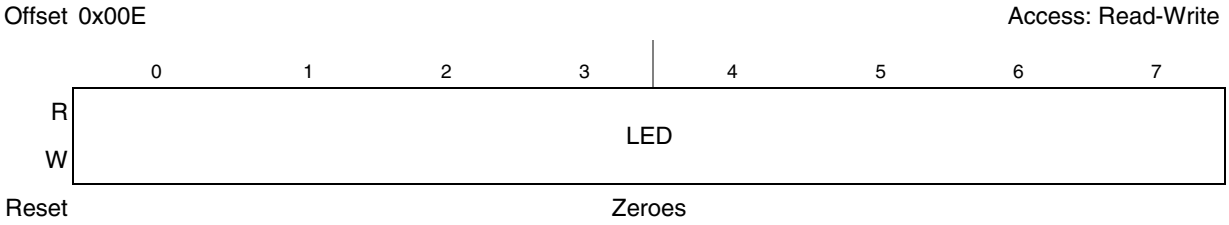


Figure 5-15. LED Register

Table 5-19. LED Register Field Descriptions

Bits	Name	Description
0-7	LED	LED[0:5] Status Control: 0 LED M[n] is off 1 LED M[n] is on.

5.3.11 I2C Block Register (I2C_BLK)

The I2C_BLK register is used to define which block of 256 registers are accessible through the I2C interface. Normally, accessing >256 bytes over I2C requires using 2-byte addressing, reducing the data rate significantly. Instead, this register serves as the upper address for all I2C transactions, allowing 1-byte addressing to be used. Since the most common registers are located in the lowest 256 locations, this enables future expansion without impacting current performance.

To ensure that I2C controllers can always access the I2C_BLK register, it has a special addressing pattern: 0xX0F. That is, it appears at offset 0x0F within the address space (0x00F, 0x10F, etc.), no matter what value is programmed in I2C_BLK.

This register does not affect the access of registers through IFC in any way.

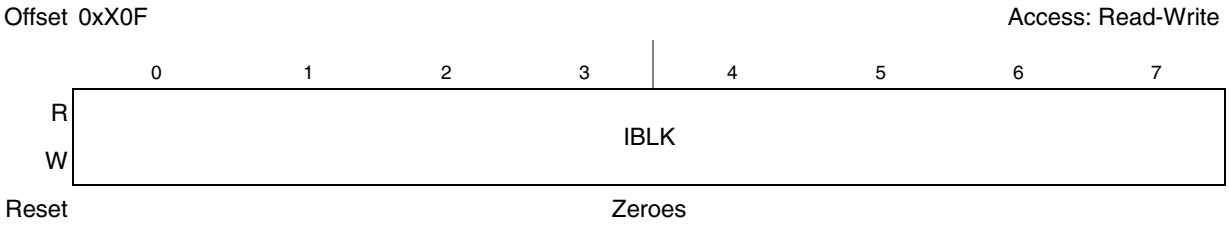


Figure 5-16. I2C_BLK Register

Table 5-20. I2C_BLK Register Field Descriptions

Bits	Name	Description
0-7	IBLK	Upper 8 address bits of I2C-register space address.

5.4 Reconfiguration/DCM Registers

This block of registers control the operation of the reconfiguration sequencer, which is used to shmoo the DUT across various voltages and frequencies, or reboot into a different flash sector. It also provides access to the DCM, which collects data during reconfiguration

NOTE

Shmooing refers to the process of varying the conditions and inputs, such as voltage, frequency, and temperature, of a system.

5.4.1 Reconfig Control Register (RCFG_CTL)

The RCFG_CTL register may be used to control the reconfiguration sequencer.

Offset 0x010 Access: Read-Write

	0	1	2	3	4	5	6	7
R W	SAM	UP	LIVE	LOCK	WDEN	—	RPWR	GO
Reset	Normal: 0010_0000 TestMode: 0000_0000							

Figure 5-17. RCFG_CTL Register

Table 5-21. RCFG_CTL Register Field Descriptions

Bits	Name	Description
0	SAM	Re-Sampling: 0 BRDCFG/DUTCFG registers are managed by the reconfiguration logic (i.e. normally). 1 On the 0->1 transition, all BRDCFG/DUTCFG registers are initialized to defaults (switches or hard-coded values).
1	UP	Update: 0 Shadow registers unaffected. 1 On the 0->1 transition, copy all BRDCFG/DUTCFG registers to their respective shadow registers. This has the effect of changing all non-reset-sampled configuration controls immediately.
2	LIVE	Immediate changes for BRDCFG registers: 0 BRDCFG registers outputs are not updated until GO or UP are changed. 1 BRDCFG registers control outputs immediately. DUTCFG registers are not affected. LIVE defaults to 1 for application mode, and 0 for test mode.
3	LOCK	Lock BRDCFG/DUTCFG registers: 0 BRDCFG/DUTCFG registers can be updated for various reasons. 1 BRDCFG/DUTCFG registers cannot be updated.
4	WDEN	Watchdog Enable: 0 The watchdog is not enabled during reconfiguration. 1 The watchdog is enabled during reconfiguration. If not disabled within 2^29 clock cycles (> 8 minutes), the system is reset. **Note:** This is not a highly-secure watchdog; software can reset this bit at any time and disable the watchdog.

Table 5-21. RCFG_CTL Register Field Descriptions (continued)

Bits	Name	Description
5	—	Reserved.
6	RPWR	Reconfiguration Power: 0 Reconfiguration does not require power cycling. 1 Reconfiguration includes power cycling.
7	GO	Reconfiguration Start: 0 Reconfiguration sequencer is idle. 1 On the 0->1 transition, the reconfiguration process begins.

NOTE

The SAM, UP, and GO fields are all edge-triggered. The SAM, UP, and GO events occur when there is a transition from 0 to 1. To re-trigger the event, the field must be set to 0, then 1.

5.4.2 Reconfig Status Register (RCFG_STAT)

The RCFG_STAT register may be used to monitor the reconfiguration sequencer activity.

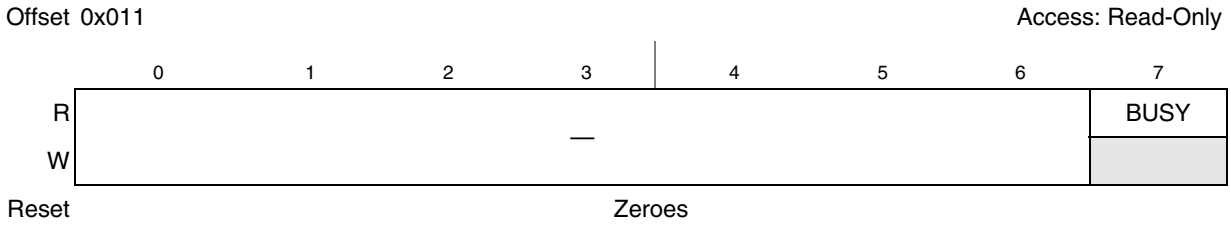


Figure 5-18. RCFG_STAT Register

Table 5-22. RCFG_STAT Register Field Descriptions

Bits	Name	Description
0-6	—	Reserved.
7	BUSY	Reconfiguration Status: 0 The system has been configured as normal. 1 The system has been reconfigured by software.

5.4.3 DCM Address Register (DCM_ADDR)

The DCM_ADDR register is used to index a location in the 256B shared SRAM. Once set to a value, the corresponding SRAM entry can be read from or written to, through the DCM_DATA register.

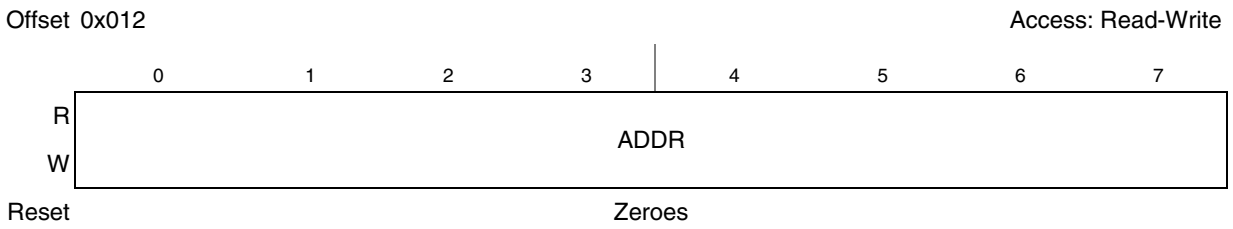


Figure 5-19. DCM_ADDR Register

Table 5-23. DCM_ADDR Register Field Descriptions

Bits	Name	Description
0-7	ADDR	Address selection for shared SRAM.

NOTE

This register is an alias of the Auxiliary SRAM Address Register (AUX_ADDR), and is provided here for backward-compatibility with the previous Design System FPGA (DS) software.

5.4.4 DCM Data Register (DCM_DATA)

The DCM_DATA register is used to read or write data to the DCM shared SRAM through an address set in the DCM_ADDR register. The standard DCM application software uses DCM_MSG as an index into the shared SRAM, but other uses are possible.

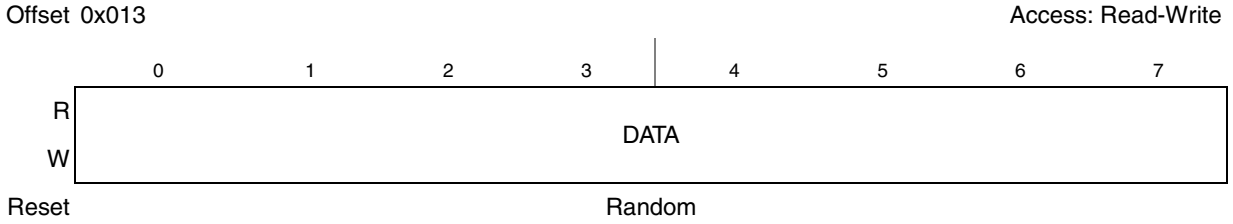


Figure 5-20. DCM_DATA Register

Table 5-24. DCM_DATA Register Field Descriptions

Bits	Name	Description
0-7	DATA	Shared SRAM data, as indexed by DCM_ADDR.

NOTE

If the DCM_CMD[CMD] register bit is set, the SRAM is locked and cannot be accessed. Writes are ignored and reads return random values.

NOTE

This register is an alias of the Auxiliary SRAM Data Register (AUX_DATA), and is provided here for backward-compatibility with the DS software.

5.4.5 DCM Command Register (DCM_CMD)

The DCM_CMD register is used to start a command by the DCM module. When the CMD bit is set, the DCM is interrupted and the command is processed. The application software can determine where the command is actually stored:

- in the XCMD field,
- in the DCM_MSG register, or
- in the shared SRAM

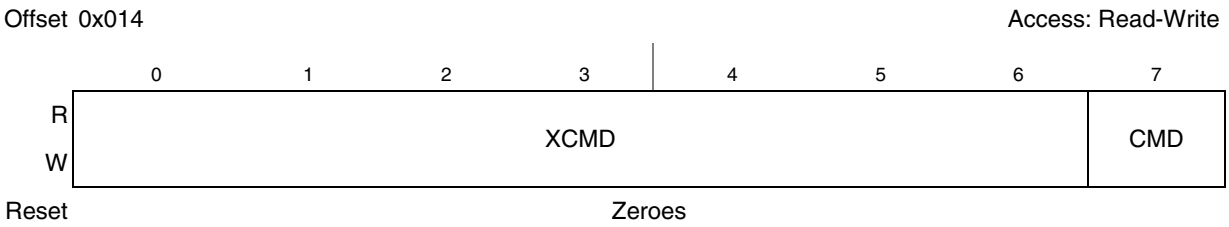


Figure 5-21. DCM_CMD Register

Table 5-25. DCM_CMD Register Field Descriptions

Bits	Name	Description
0-6	XCMD	Extended Command codes. 0 if unused.
7	CMD	Command Start: 0 No command 1 Start indicated command. When CMD=1, the shared SRAM is locked by the DCM and is not accessible by other means.

5.4.6 DCM Message Register (DCM_MSG)

The DCM_MSG register is used to send a message to the DCM module. The standard DCM application software uses DCM_MSG as an index into the shared SRAM, but other uses are possible.

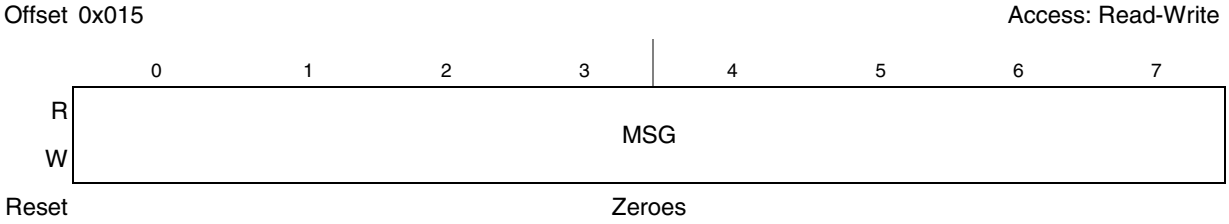


Figure 5-22. DCM_MSG Register

Table 5-26. DCM_MSG Register Field Descriptions

Bits	Name	Description
0-7	MSG	Message to DCM software. Address at which the message to be processed, is stored in shared SRAM.

5.4.7 Debug Control Register (GDC)

The GCD register is used to select internal registers of the DCM module for read or write access.

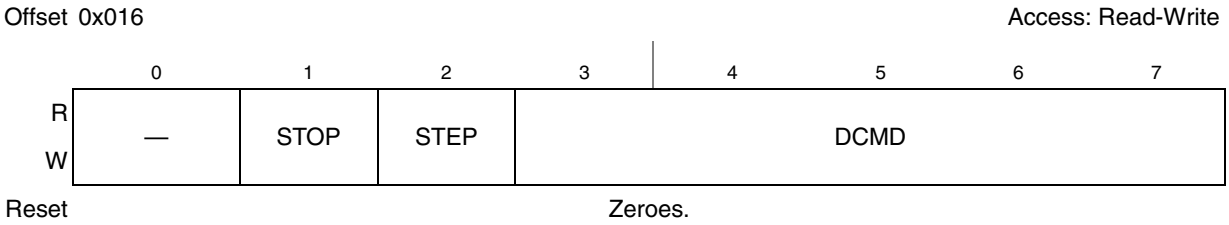


Figure 5-23. GDC Register

Table 5-27. GDC Register Field Descriptions

Bits	Name	Description
0	—	Reserved.
1	STOP	Stop processor: 0 Processor runs normally. 1 Processor halts on the completion of the current instruction.

Table 5-27. GDC Register Field Descriptions (continued)

Bits	Name	Description
2	STEP	Step processor: 0 No effect. 1 On 0->1 transition, execute the current instruction and then return to the stop state.
3-7	DCMD	Select the following resource for RW or RO access (as indicated) by GDD: 00000:ROGS: general status, 8 bits comprising: 7-5: 000 4 : STOP: set if CPU is halted 3 : IPLA: set if IPL_LOADER is running 2 : BYP : set if OCM is bypassed (disabled) 1 : GFLT: set if GMSA faulted (bad opcode) 0 : IFLT: set if IPL_LOADER faulted (erased EEPROM). 00001: ROPC[15:8], program counter high 00010: ROPC[7:0], program counter low 00011: RWProgram memory, as indexed by VA 00100: ROSP[15:8], stack pointer high 00101: ROSP[7:0], stack pointer low 00110: ROopcode at PC 00111: ROSR: CPU status register, consisting of: 7-4: 0000 3 : IQ: interrupt request pin 2 : IM: interrupt mask 1 : CO: carry out 0 : C: carry 01000: ROLR[15:8], link register high 01001: ROLR[7:0], link register low 01010: ROMR[15:8], memory register high 01011: ROMR[7:0], memory register low 01100: ROMU[15:8], memory user register high 01101:ROMU[7:0], memory user register low 01110: RWVA[15:8], virtual address register 01111: RWVA[7:0], virtual address register 10000: RWBP[15:8], virtual address register 10001: RWBP[7:0], virtual address register 10010: ROIPL[15:8] (bootloader) current address 10011: ROIPL[7:0] (bootloader) current address 10100: ROIPL_HINT: MSB of address with last defined program byte. LSB is always 0xFF. 10101: ROTOS (top of stack; last instruction result) 10110: ROGPI1: GPI1 input port. 10111: ROGPI2: GPI2 input port. 11000: ROGPO1: GPO1 output port. 11001: ROGPO2: GPO2 output port. 11010: ROIVEC: Current interrupt vector asserted. 11111: ROVER: GMSA debugger version. All other values are reserved.

NOTE

QIXIS uses GMSA core V2, while PIXIS uses GMAS core V1. The GDC and corresponding DCMD codes are different as compared to those devices. Software can compatibly query DCMD[ID=0x1F] to get the version number:

– 0x00 - GMSA V1 debugger.
– 0x01 - GMSA V2 debugger.

For details, see the GMSA specification or the DINK source code.

5.4.8 DCM Debug Data Register (GDD)

The GDD register is used to read or write the data selected by the GDC register, in the DCM module. For more information, see Section 5.4.7, “Debug Control Register (GDC)”.

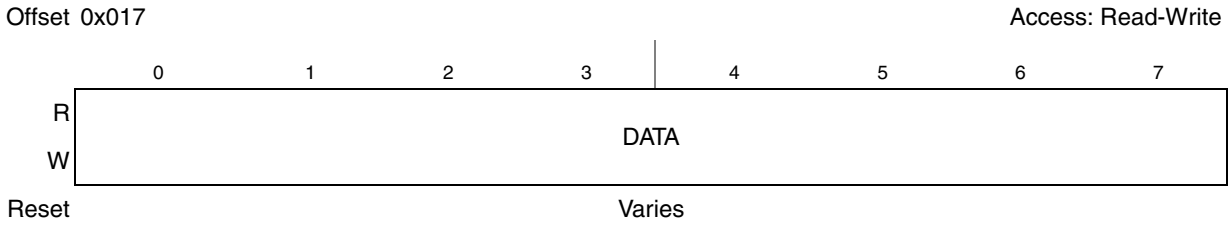


Figure 5-24. GDD Register

Table 5-28. GDD Register Field Descriptions

Bits	Name	Description
0-7	DATA	Read: Access data as described by GDC. Write: Modify data as described by GDC (where possible).

5.4.9 DCM Message Acknowledge Register (DCM_MACK)

The DCM_MACK register stores the response to the last message sent to the DCM processor through DCM_CMD/DCM_MSG registers.

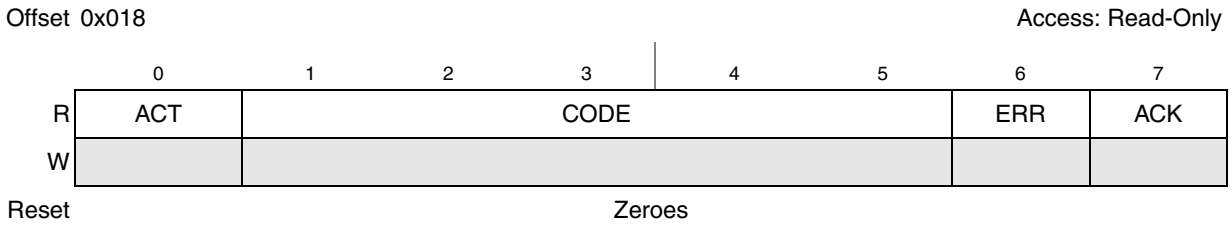


Figure 5-25. DCM_MACK Register

Table 5-29. DCM_MACK Register Field Descriptions

Bits	Name	Description
0	ACT	Activity Indicator: 0 No activity. 1 Background data collection is occurring.
1-5	CODE	Response Code (optional): 0, unless application software specifies otherwise.
6	ERR	Message Ack Error Code: 0 No error. 1 Error occurred. An error code might be present in CODE (application software-dependant).
7	ACK	Message Ack: 0 Command not completed; or no command ever received.

5.4.10 Watchdog Register (RCFG_WATCH)

The RCFG_WATCH register selects the watchdog timer value used during reconfiguration processes. When enabled by RCFG_CTL[WDEN], the watchdog timer begins to count down. If the DUT software has not disabled or restarted the watchdog timer within the specified limit, the system will be restarted.

Note that the watchdog timer is not conditional upon a reconfiguration sequence being active. While it is enabled along with RCFG_CTL[GO] as part of a reconfiguration sequence; in fact, it is independent and can be enabled for any reason.

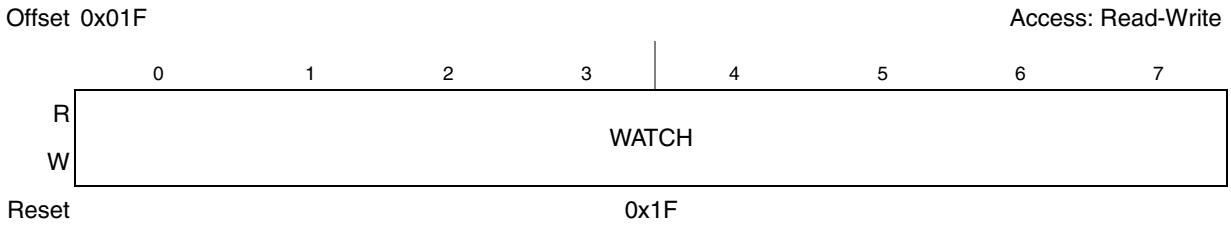


Figure 5-26. RCFG_WATCH Register

Table 5-30. RCFG_WATCH Register Field Descriptions

Bits	Name	Description
0-7	WATCH	Watchdog timer value.

The watchdog time-out time is determined by the formula:

```
time-out = [ WATCH x (2.01326592sec) ] + 2.01326592sec.
```

Some examples values for PX_WATCH register values:

Table 5-31. Watchdog Timer Values

Time-out Value		Time-out
Binary	Hex	
11111111	0xFF	8.59 min
01111111	0x7F	4.29 min
00111111	0x3F	2.15 min
00011111	0x1F	1.07 min
00001111	0x0F	32.1 sec
00000111	0x07	16.1 sec
00000011	0x03	8.05 sec
00000001	0x01	4.027 sec
00000000	0x00	2.013 sec

NOTE

The watchdog value cannot be changed while the watchdog timer is operating (that is, while `RCFG_CTL[WDEN] = 1`).

5.5 Power Control/Status Registers

The power registers provide the ability to monitor general power status, as well as individual power status (for those supplies that have reporting capability). Other registers provide limited power control features (most power control is through the PMBus/I2C interface).

5.5.1 Power Control Register (PWR_CTL1)

The PWR_CTL1 register is used to control some of the power supplies. There are not many, as most supplies are enabled or disabled by normal sequencing (power switches, reconfiguration logic).

Offset PWR_CTL1: 0x020 Access: Read-Write

	0	1	2	3	4	5	6	7
R W	VID_EN	VID_FRC	—	rsv	rsv	rsv	VID0	VID1
Reset	Varies.							

Figure 5-27. PWR_CTL1 Registers

Table 5-32. PWR_CTL1 Register Field Descriptions

Bits	Name	Description
0	VID_EN	VID Encoding Mode 0 Assume processor is not driving VID; ignore the signals. (default) 1 Use VID to set core voltage. (not used in Version -01)
1	VID_FRC	VID Encoding Force 0 Use processor-supplies VID encoding, if any. 1 Force value in PWR_CTL1[VID[0:n]] instead of hardware signals.
2	—	Reserved.

5.5.2 Power Control Register (PWR_CTL2)

The PWR_CTL2 register is used by application software to force the system to power off.

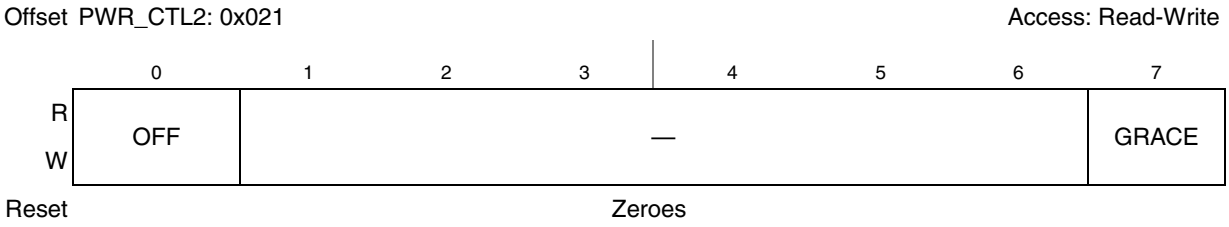


Figure 5-28. PWR_CTL2 Registers

Table 5-33. PWR_CTL2 Register Field Descriptions

Bits	Name	Description
0	OFF	Force Power Off: 0 No action. 1 On 0-to-1 transition, the system power supply is forced to turn off. The bit must be reset to zero before additional power cycles can occur.
1-6	—	Reserved.
7	GRACE	Graceful Power Off: 0 No action. Power switch turns off power immediately 1 When the external power switch is pressed and the system is on, assert IRQ7, wait 3 seconds, then power down normally. This allows software to sync disks or perform other cleanup activities before powering down. **Note:** This bit is cleared on power-up. **Note:** Software-initiated power cycles are not affected by this setting.

5.5.3 Power Main Status Register (PWR_MSTAT)

The PWR_MSTAT register monitors the overall power status of the board, including that of the ATX (or bench) power supply used to power all other rails.

Offset PWR_MSTAT: 0x024 Access: Read-Only

	0	1	2	3	4	5	6	7
R	ATXON	ATXGD	SYS_PG	DUT_PG	PWROK	—	S_STATE	
W								

Reset Varies.

Figure 5-29. PWR_MSTAT Registers

Table 5-34. PWR_MSTAT Register Field Descriptions

Bits	Name	Description
0	ATXON	ATX Power Supply Control Status: 0 Power supply is set to off. 1 Power supply is set to on.
1	ATXGD	ATX Power Supply Status: 0 Power supply is off or not yet stable. 1 Power supply is on and stable.
2	SYSPG	System Power Supply Status (on board voltage rails before DUT power on). 0 One or more power supplies are off or not yet stable. 1 All power supplies are on and stable. Sampled output of system voltage monitor U17
3	DUT_PG	DUT Power Supply Status (DUT voltage rails monitoring). 0 One or more power supplies are off or not yet stable. 1 All power supplies are on and stable. Sampled output of system voltage monitor U20
4	PWROK	General Power Status: 0 One or more power supplies are off or not yet stable. 1 All power supplies are on and stable.
5-7	—	Reserved.

5.5.4 Power Status Registers (PWR_STATn)

The PWR_STAT*n* registers are used to monitor the status of individual power supplies. If a bit is set to 1, the respective power supply is operating correctly.

Note that unassigned bits default to one, allowing power failure detection to be easily performed (if the value is not 0xFF, at least one supply is not operating).

Due to the high variability of hardware devices used, PWR_STATn register bits are not assigned any universally fixed values. For more details, see the target platform documentation. Table 5-36 provides a list of customary assignments, but note that it is only for reference.

Offset PWR_STAT1: 0x025
PWR_STAT2: 0x026

Access: Read-Only

	0	1	2	3	4	5	6	7
R	PWR*n*.0	PWR*n*.1	PWR*n*.2	PWR*n*.3	PWR*n*.4	PWR*n*.5	PWR*n*.6	PWR*n*.7
W								
Reset	Ones if all working correctly.							

Figure 5-30. PWR_STAT*n* Registers

Table 5-35. PWR_STAT*n* Register Field Descriptions

Bits	Name	Description
0-7	PWR*n*.b	Power Supply Status: 0 Power supply is off or is out of limits. 1 Power supply at correct levels.

Table 5-36. PWR_STAT*n* Customary Bit Assignment

Reset Force	Customary Use	Description
PWR1.0	CVDD	SDHC/USB supply voltage. Sampled DUT_Power voltage monitor (U20.10) - CVDD_OK
PWR1.1	SVDD	SERDES core voltage 9132 QDS: Unused, reads as '1'.
PWR1.2	XPADVDD	SERDES transceiver voltage. Sampled DUT_Power voltage monitor (U20.10) - XPADVDD_OK,
PWR1.[3-5]	-	9132 QDS: Unused, reads as '1'.
PWR1.6	VDD	Maple/Titanium power. Sampled DUT_Power voltage monitor (U20.11) - VDD_OK,
PWR1.7	VDDC	Platform and Core power voltage Sampled Zilker Core PS (U46) VCORE_PWR_OK signal
PWR2.0	POVDD	Fuse/Test power 9132 QDS: Unused, reads as '1'.
PWR2.1	BVDD	IFC bus Sampled DUT_Power voltage monitor (U20.15) - BVDD_OK
PWR2.2	X1VDD	ANT2/3 interface power Sampled DUT_Power voltage monitor (U20.8) - X1VDD_OK
PWR2.3	X2VDD	ANT2/3 interface power Sampled DUT_Power voltage monitor (U20.9) - X1VDD_OK
PWR2.4	LVDD	TSEC interface power Sampled DUT_Power voltage monitor (U20.17) - LVDD_OK,
PWR2.5	OVDD	Miscellaneous signalling power. Sampled DUT_Power voltage monitor (U20.18) - OVDD_OK
PWR2.6	G2VDD	DDR1 power Sampled DDR1 PS (U84) PS2_DDR_PG_L output
PWR2.7	G1VDD	DDR1 power Sampled DDR1 PS (U21) PS1_DDR_PG_L output

Note that the bit assignment table is not mandatory; some architectures may have radically different reset requirements.

5.6 Clock Control Registers

The clock control registers control and monitor the 4 primary high-speed clock inputs used with Power Architecture processors:

- SYSCLK
- DDR1CLK
- DDR2CLK
- DSPCLK

Note there may be other clock inputs, such as SERDES, RTC, and Ethernet. But these are fixed or are limited in range and controlled by methods, such as BRDCFG or software.

Also note that the current implementation of these registers directly mirrors the hardware usage of the IDT307 clock synthesizer. If the hardware changes, the registers must also change.

5.6.1 Speed Register (CLK_SPD1)

The CLK_SPD1 register is used to report the user-selectable speed settings (from switches) for the SYSCLK and DSPCLK clock.

Values in the CLK_SPD1 register are used by boot software to accurately initialize timing-dependant parameters, such as those for UART baud rates, I2C clock rates, and DDR memory timing parameters.

Offset 0x030 Access: Read-Only

	0	1	2	3	4	5	6	7
R	0	0	SW_DSP[0:1]		0	0	SW_SYS[0:1]	
W								-
Reset	Sampled from user-supplied switches.							

Figure 5-31. CLK_SPD1 Register

Table 5-37. CLK_SPD1 Register Field Descriptions

Bits	Name	Description
0-3	SW_DSP	Reflects switch SW_CFG_D2_DDRCLK_SYNTH[0:1] settings used to preset DSPCLK DSP_CLK in frequency selection (Sampled from switches SW_DSP_FS[0:1]) 0066.666667 MHz 01100.000000 MHz 10133.333333 MHz 11160.000000 MHz
4-7	SW_DSP	Reflects switch SW_CFG_SYSCLK_SYNTH[0:1] settings used to preset SYSCLK.

5.6.2 Speed Register (CLK_SPD2)

The CLK_SPD2 register is used to report the user-selectable speed settings (from switches) for the DDR1 and DDR2 PSC9132 input clocks.

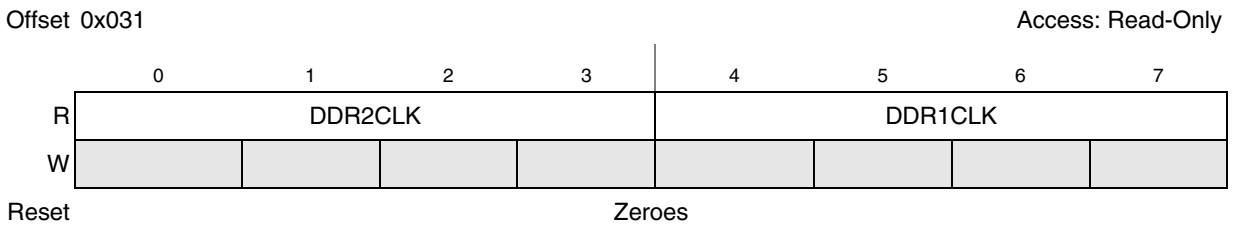


Figure 5-32. CLK_SPD2 Register

Table 5-38. CLK_SPD2 Register Field Descriptions

Bits	Name	Description
0-3	DDR2CLK	Reflects switch SW_CFG_D2_DDRCLK_SYNTH[0:1] settings used to preset DDR2CLK.
4-7	DDR1CLK	Reflects switch SW_CFG_D1_DDRCLK_SYNTH[0:1] settings used to preset DDR1CLK.

5.6.3 Speed Register (CLK_CTL)

The CLK_CTL register is used to control the behaviour of the clock setup subsystem.

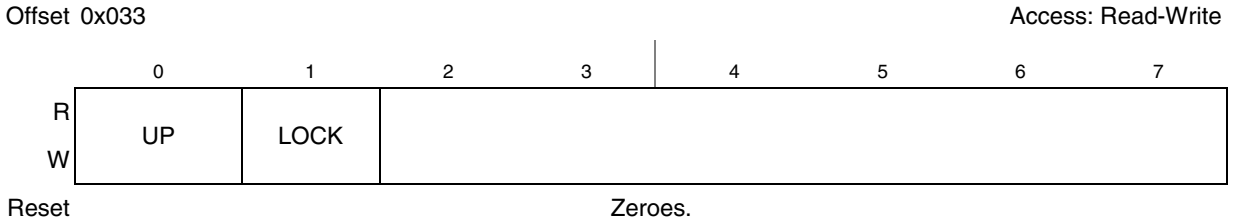


Figure 5-33. CLK_CTL Register

Table 5-39. CLK_CTL Register Field Descriptions

Bits	Name	Description
0	UP	Clock config value updating 0 Reset sequencer operates normally. 1 On the 0->1 transition, update (reprogram) all programmable clocks from registers **WARNING:** Devices (the DUT) may fail if the updated frequencies are not within its PLL lock tolerances.
1	LOCK	Lock Clock Data 0 Normal operation: **QIXIS** uses manual SW values for clock configuration data. 1 Locked: **QIXIS** will not update the clock from manual software values. This is used when software wants to program its non-standardized values.
2-7	—	Reserved

5.6.4 Clock Configuration Registers (CLK_SYSDn)

The clock registers define the 24-bit pattern used to initialize the IDT307 SYSCLK generator.

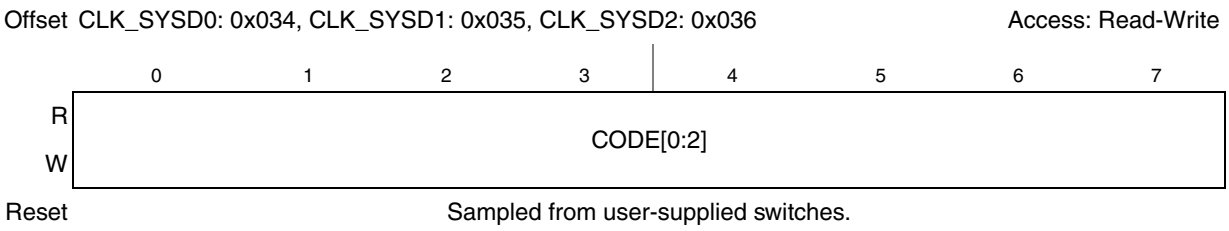


Figure 5-34. CLK_SYSD*n* Registers

Table 5-40. CLK_SYSD*n* Register Field Descriptions

Bits	Name	Description
0-7	CODE	Contains a 24-bit pattern used to program an IDT307 with a frequency value ranging from 0.5MHz to 200 MHz. CLK_xxxD0 contains the MSB through CLK_xxxD2, which contains the LSB.

The initial content of each register is mapped from switches on the board. Software can change it later to get finer resolution, if needed.

NOTE

These registers might change with newer platforms.

5.6.5 Clock Configuration Registers (CLK_DSPDn)

The clock registers define the 24-bit pattern used to initialize the IDT307 DSPCLKK generator.

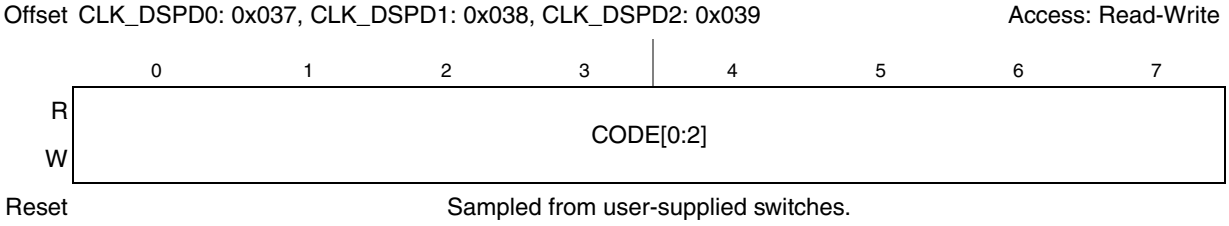


Figure 5-35. CLK_DSPD*n* Registers

Table 5-41. CLK_DSPD*n* Register Field Descriptions

Bits	Name	Description
0-7	CODE	Contains a 24-bit pattern used to program an IDT307 with a frequency value ranging from 0.5MHz to 200 MHz. CLK_xxxD0 contains the MSB through CLK_xxxD2, which contains the LSB.

The initial content of each register is mapped from switches on the board. Software can change it later to get finer resolution, if needed.

NOTE

These registers might change with newer platforms.

5.6.6 Clock Configuration Registers (CLK_DDR1Dn)

The clock registers define the 24-bit pattern used to initialize the IDT307 DSPCLKK generator.

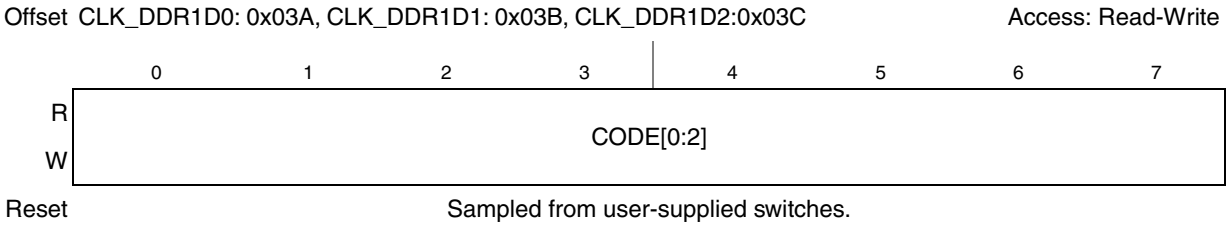


Figure 5-36. CLK_DDR1D*n* Registers

Table 5-42. CLK_DDR1D*n* Register Field Descriptions

Bits	Name	Description
0-7	CODE	Contains a 24-bit pattern used to program an IDT307 with a frequency value ranging from 0.5MHz to 200 MHz. CLK_xxxD0 contains the MSB through CLK_xxxD2, which contains the LSB.

The initial content of each register is mapped from switches on the board. Software can change it later to get finer resolution, if needed.

NOTE

These registers might change with newer platforms.

5.6.7 Clock Configuration Registers (CLK_DDR2Dn)

The clock registers define the 24-bit pattern used to initialize the IDT307 DDR2CLKK generator.

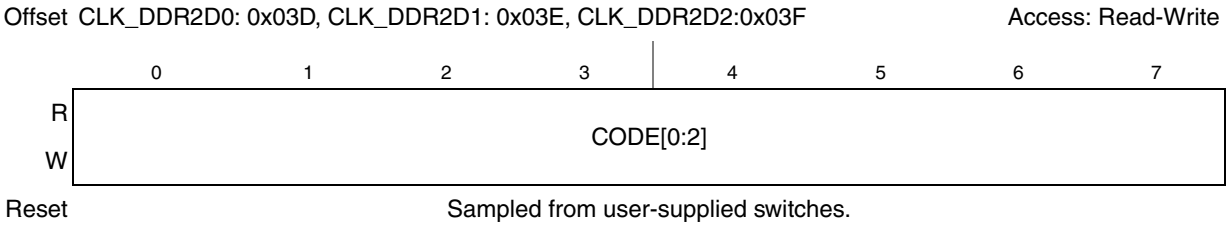


Figure 5-37. CLK_DDR2D*n* Registers

Table 5-43. CLK_DDR2D*n* Register Field Descriptions

Bits	Name	Description
0-7	CODE	Contains a 24-bit pattern used to program an IDT307 with a frequency value ranging from 0.5MHz to 200 MHz. CLK_xxxD0 contains the MSB through CLK_xxxD2, which contains the LSB.

The initial content of each register is mapped from switches on the board. Software can change it later to get finer resolution, if needed.

NOTE

These registers might change with newer platforms.

5.7 Reset Control Registers

The reset control register group handles reset behaviour configuration and general monitoring of resets.

5.7.1 Reset Control Register (RST_CTL)

The RST_CTL register is used configure or trigger reset actions.

Offset 0x040 Access: Read-Write

	0	1	2	3	4	5	6	7
R	RST	WAIT	—				REQMD	
W								

Reset WAIT and REQMD are set from switches; rest are zeroed.

Figure 5-38. RST_CTL Register

Table 5-44. RST_CTL Register Field Descriptions

Bits	Name	Description
0	RST	Reset Trigger 0 No action. 1 On 0->1 transition, (re)start the reset sequence. The bit must be reset to 0 to trigger additional times.
1	WAIT	Reset Sequencer Wait 0 Reset sequencer operates normally. 1 Upon any reset source, enter the special state RMT-WAIT and wait for a software remote release action through RST_CTL[RST].
2-5	—	Reserved.
6-7	REQMD	RESET_REQ_B assertion handling: 00 Disabled: Do nothing. 01 Loopback: Assert HRESET_B to DUT, but do not start reset sequence. 10 Not applicable 11 Normal: Assert PORESET_B to DUT to start normal reset sequence.

5.7.2 Reset Status Register (RST_STAT)

The RST_STAT register reports the current status of various reset-related signals.

Offset 0x041 Access: Read-Only

	0	1	2	3	4	5	6	7
R	WAIT	—				SRST	HRST	RREQ
W								

Reset Zeroes, unless signal is asserted.

Figure 5-39. RST_STAT Register

Table 5-45. RST_STAT Register Field Descriptions

Bits	Name	Description
0	WAIT	Reset Waiting 0 Reset sequencer is operating normally. 1 Reset sequencer is in RMT-WAIT state, waiting for permission to proceed.
1-4	—	Reserved.
5	SRST	SRESET_B signal: (not used in current version) 0 SRESET_B is not asserted. 1 SRESET_B is asserted.
6	HRST	HRESET_B signal: 0 HRESET_B is not asserted. 1 HRESET_B is asserted.
7	RREQ	RESET_REQ_B signal: 0 RESET_REQ_B is not asserted. 1 RESET_REQ_B is asserted.

Note that on older platforms, the following mappings might be used:

- SRESET_B=> HRESET_B
- HRESET_B=> PORESET_B
- HRESET_REQ_B=> RESET_REQ_B

5.7.3 Reset Reason Register (RST_REASON)

The RST_REASON register is used to report the cause of the most-recent reset cycle.

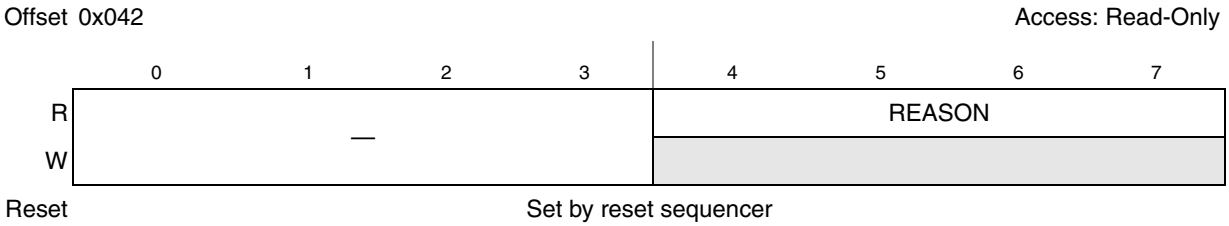


Figure 5-40. RST_REASON Register

Table 5-46. RST_REASON Register Field Descriptions

Bits	Name	Description
0-3	—	Reserved.
4-7	REASON	The field will contain one of the following codes, indicating the target system was reset for that reason: 0000 - power-on reset 0001 - COP/JTAG HRESET_B was asserted 0010 - Watchdog timer (during reconfiguration) 0011 - RST_CTL[RST] was set 0100 - Reset switch (chassis or on-board) was pushed. 0101 - RCFG_CTL[GO] (i.e. reconfiguration reset) was asserted. 0110 - RESET_REQ_B assertion (from processor) was asserted. 0111 - EONCE/JTAG HRESET_B was asserted.

5.7.4 Reset Force Registers (RST_FORCEn)

The RST_FORCE*n* registers are used to force reset to a particular device, independent of the general reset sequencer. As long as a bit is set to 1, the reset signal to grouped devices will be asserted.

Due to the high variability of hardware devices used, RST_FORCE*n* registers do not have a universally-defined purpose. See the target platform documentation for details.Table 5-48 lists customary assignments, but keep in mind it is only for reference.

Offset RST_FORCE1: 0x043
RST_FORCE2: 0x044
RST_FORCE3: 0x045

Access: Read-Write

	0	1	2	3	4	5	6	7
R W	RST*n*.0	RST*n*.1	RST*n*.2	RST*n*.3	RST*n*4	RST*n*.5	RST*n*.6	RST*n*.7
Reset	Zeroes							

Figure 5-41. RST_FORCE*n* Registers

Table 5-47. RST_FORCE*n* Register Field Descriptions

Bits	Name	Description
0-7	RST*n*.b	Specified Reset signal, where n - Reset Register number b - number of bit position, respectively

Table 5-48. Reset Force Register Change History

Reset Force	Customary Use	Description
RST1.0	General Board Reset	SGMII PHY[1:2],USB PHY,1588 Card,I2C Switches,LB Nor Flash, SPI and SDHC Mux EN, SPI1 and UART0 Mux EN, RS-232 PHY OFF, PCIe SLOT
RST1.1	DDR1_RST	DDR1 memory devices
RST1.2	DDR2_RST	DDR2 memory devices
RST1.3	RF1_CARD	RF1 ADI module
RST1.4	RF2_CARD	RF2 ADI module

Table 5-48. Reset Force Register Change History (continued)

RST1.5	RF3_CARD	RF3 ADI module
RST1.6	SLIC1	SLIC1 on board PHY
RST1.7	SLIC2.	SLIC2 on board PHY
RST2.0	TPR_RESET	Test Port Reset
RST2.1	GPS_RST	GPS Module Reset
RST2.2	DUT_CLKIN_EN	DUT Clocks input Enable
RST2.3	TPR_GPIO32	Auxiliary Test Port IO Reserved
RST2.[4:5]	—	Reserved
RST2.6	SGMII_PHY1_RST	Reset of SGMII PHY1 0 PHY1_SRESET_B is not asserted 1 PHY1_SRESET_B is asserted
RST2.7	SGMII_PHY2_RST	Reset of SGMII PHY2 0 PHY2_SRESET_B is not asserted 1 PHY2_SRESET_B is asserted
RST3.0	DUT_HRESET_FORCE	DUT_HRESET_FORCE: 0 DUT_HRESET_B is not asserted 1 DUT_HRESET_B is asserted
RST3.1	DUT_SRESET_FORCE	DUT_SRESET_FORCE: 0 DUT_SRESET_B is not asserted 1 DUT_SRESET_B is asserted
RST3.2	EP_RCn	EP_RCn signal controls the SD1_PEFCLK source When PCI Card is present, (rst_force3[2] == 1) EP_RCn signal goes high and changes SD1_REFCLK clock source from internal to external (PCIe slot).
RST3.3	ATX_PWR_FORCE	ATX_PWR_FORCE signal: The 0->1 transition forces "power on" signal of board external power supply (ATX). This signal overrides on board "Power ON" push button. By default, this bit is 0.
RST3.4	SLAVE_HRESET_FORCE	SLAVE_HRESET_FORCE: 0 DUT_SRESET_B is not asserted 1 DUT_SRESET_B is asserted This signal is enabled when on board SW12[3] manual switch is in state OFF.
RST3[5]	TMP_DETECT	TMP_DETECT signal 0 TMP_DETECT is not asserted, (default) 1 TMP_DETECT is asserted
RST3[6]		Reserved.
RST3.7	SLAVE_ATX_PWR_FORCE	SLAVE_ATX_PWR_FORCE: On 0->1->0 transition is force of powering ATX PS of slave board. This signal emulates of on board power button (toggle ON/OFF). By default this bit is 0 This functional available when SPARE[1] = 1'b1 (SW12[3] = OFF)

Table 5-48. Reset Force Register Change History (continued)

RST2.6	-SGMII_PHY1_RST	Reset of SGMII PHY1 0 PHY1_SRESET_B is not asserted 1 PHY1_SRESET_B is asserted
RST2.7	-SGMII_PHY2_RST	Reset of SGMII PHY2 0 PHY2_SRESET_B is not asserted 1 PHY2_SRESET_B is asserted

5.8 Board Configuration Registers

This block of registers control the configuration of the board. BRDCFG registers are always static, driven at all times power is available. There are up to 16 registers providing up to 128 control options; however, not every platform implements all the registers.

Note that BRDCFG registers are often used to demultiplex the DUT, which means their definition is highly variable. To facilitate reuse, the lower-numbered registers are used to control cross-platform configurations (such as LBMAP) which are constant across all platforms, while higher-numbered registers might be unique for each board.

5.8.1 Board Configuration Register 0 (BRDCFG0)

The BRDCFG0 register is used to control the local-bus mapping for the target system.

Offset 0x050 Access: Read-Write

	0	1	2	3	4	5	6	7
R	NAND_WB	NOR_WB	VBANK[0:2]			LBMAP[0:2]		
W								
Reset	Sampled from switches.							

Figure 5-42. BRDCFG0 Register

Table 5-49. BRDCFG0 Register Field Descriptions

Bits	Name	Description
0	NAND_WB	IFC bus width selection for NandFlash on boards device. 0 8 bit (default) 1 16 bit Sampled from on board manual switch
1	NOR_WB	IFC bus width selection for NorFlash on board device. 0 8 bit 1 16 bit (default) Sampled from on board manual switch

Table 5-49. BRDCFG0 Register Field Descriptions (continued)

Bits	Name	Description
2-4	VBANK[0:2]	NOR Flash Memory Bank select (accept LBMAP[0:2] value when LBMAP[0]=0 - NorFlash boot
5-7	LBMAP[0:2]	Route the LCS[0:2] signals to various target devices. For access to NOR, Nand flash, PromJet, and SRAM_DIMM, LBMAP includes virtual banking support. 000NOR=CS0;Nand=CS1;QIXIS = CS2 100NOR=CS1;Nand=CS0;QIXIS = CS2 (default) 111PJET=CS0;Nand=CS1;QIXIS,Nor=CS2 101LS1 = SAM_DIMM, QIXIS = CS2 Sampled from on board manual switch

Note that while LBMAP is standard feature of DS systems, the variability of devices used on the local bus means that a board-dependant table is used. See the relevant board documentation for details.

5.8.2 Board Configuration Register 1 (BRDCFG1)

The BRDCFG1 register controls clocking configuration on the target system: SYSCLK and DSPCLK.

Offset 0x051 Access: Read-Write

	0	1	2	3	4	5	6	7
R W	DSPCLK				SYSCLK[0:3]			
Reset	Sampled from switches.							

Figure 5-43. BRDCFG1 Register

Table 5-50. BRDCFG1 Register Field Descriptions

Bits	Name	Description
0-3	DSPCLK	DSP_CLK in frequency selection (Sampled from switches SW_DSP_FS[1:0]) 00 - 66.666667 MHz 01 - 100.000000 MHz 10 - 133.333333 MHz 11 - 160.000000 MHz Other values can be overridden by software as follows: 0000 - 66.666667 MHz 0001 - 83.333 MHz 0010 - 100.000 MHz 0011 - 125.000 MHz 0100 - 133.333 MHz 0101 - 150.000 MHz 0110 - 160.000 MHz 0111 - 166.666 MHz 1000 - 75.000 MHz 1001 - 80.000 MHz 1010 - 90.000 MHz OTHERS (reserved - default to 100.00MHz).
4-7	SYSCLK	SYS_CLK in frequency selection (Sampled from switches SW_SYS_FS[1:0]) 00 - 66.666667 MHz 01 - 100.000000 MHz 10 - 133.333333 MHz 11 - 160.000000 MHz Other values can be overridden by software as follows: 0000 - 66.666667 MHz 0001 - 83.333 MHz 0010 - 100.000 MHz 0011 - 125.000 MHz 0100 - 133.333 MHz 0101 - 150.000 MHz 0110 - 160.000 MHz 0111 - 166.666 MHz 1000 - 75.000 MHz 1001 - 80.000 MHz 1010 - 90.000 MHz OTHERS (reserved - default to 100.00MHz).

5.8.3 Board Configuration Register 2 (BRDCFG2)

The BRDCFG2 register controls clocking configuration on the target system: DDR1_CLK and DDR2_CLK.

Offset 0x052 Access: Read-Write

	0	1	2	3	4	5	6	7
R	DDR2_CLK				DDR1_CLK[0:3]			
W								
Reset	Sampled from switches.							

Figure 5-44. BRDCFG2 Register

Table 5-51. BRDCFG2 Register Field Descriptions

Bits	Name	Description
0-3	DDR2CLK	D1_DDR_CLK in frequency selection (Sampled from switches SW_D1_FS[0:1]) 00 - 66.666667 MHz 01 - 100.000000 MHz 10 - 133.333333 MHz 11 - 160.000000 MHz Other values can be overridden by software as follows: 0000 - 66.666667 MHz 0001 - 83.333 MHz 0010 - 100.000 MHz 0011 - 125.000 MHz 0100 - 133.333 MHz 0101 - 150.000 MHz 0110 - 160.000 MHz 0111 - 166.666 MHz 1000 - 75.000 MHz 1001 - 80.000 MHz 1010 - 90.000 MHz OTHERS(reserved - default to 100.00MHz).
4-7	DDR1CLK	D2_DDR_CLK in frequency selection (Sampled from switches SW_D2_FS[0:1]) 00 - 66.666667 MHz 01 - 100.000000 MHz 10 - 133.333333 MHz 11 - 160.000000 MHz Other values can be overridden by software as follows: 0000 - 66.666667 MHz 0001 - 83.333 MHz 0010 - 100.000 MHz 0011 - 125.000 MHz 0100 - 133.333 MHz 0101 - 150.000 MHz 0110 - 160.000 MHz 0111 - 166.666 MHz 1000 - 75.000 MHz 1001 - 80.000 MHz 1010 - 90.000 MHz OTHERS (reserved - default to 100.00MHz).

5.8.4 Board Configuration Register 3 (BRDCFG3)

The BRDCFG3 register is used to control DUT serial interface demuxing configurations on the target system.

Offset 0x053 Access: Read-Write

	0	1	2	3	4	5	6	7
R	SPI1MUX	SPI1_FLMUX	SPI1_SLICMUX	SDHC_MUX	—			RF3_SPI2
W								
Reset	0000_0000							

Figure 5-45. BRDCFG3 Register

Table 5-52. BRDCFG3 Register Field Descriptions

Bits	Name	Description
0	SPI1MUX	SPI1 MUX control 0 SPI1 Interface (default) 1 UART3, CKSTP0_OUT_B, CKSTP1_OUT_B (Controls QIXIS FPGA_SPI1_SEL_L output)
1	SPI1_FLMUX	SPI1 FLASH MUX control 0 SPI1 is routed to on board SPI Flash memory devices (default) SPI1_CS0 Atmel RapidS Flash AT45DB081D SPI1_CS1 ST SPI Flash M95640 SPI1_CS2 DualRead W25X80VSSIG SPI1_CS2 Spansin Flash S25FL128P0XNFI00 1 SPI1 is routed to 1588 Card and SPI1_SLICMUX SPI1_CS2 l 1588 IF Card (Controls QIXIS SPI1_FLASH_SEL_B output)
2	SPI1_SLICMUX	SPI1 SLIC MUX control 0 SPI1 is routed to the SLIC1 PHY - SPI_CS0 (default) 1 SPI2 is routed to the SLIC1 PHY - SPI_CS1 (Controls QIXIS SLIC_SPI1_SEL output)
3	SDHC_MUX	SDHC MUX control 0 SDHC IF is routed to the SD Card Connector (J45) - (default) 1 USIM IF is routed to the Power Translator (U113) DMA0 channel is routed to QIXIS (Controls QIXIS SDHC_MUX_SEL_L output)
4-6	—	Reserved
7	RF3_SPI2	RF3_SPI2 control 0 SPI_CS[2:3] are routed to RF2 card- (default) 1 SPI_CS[2:3] are routed to RF3 card (Controls QIXIS RF3_SPI2_EN output) - Applicable only for BSC9132QDS Rev3 board

5.8.5 Board Configuration Register 4 (BRDCFG4)

The BRDCFG4 register controls signals multiplexing for SerDes based peripherals on the target system.

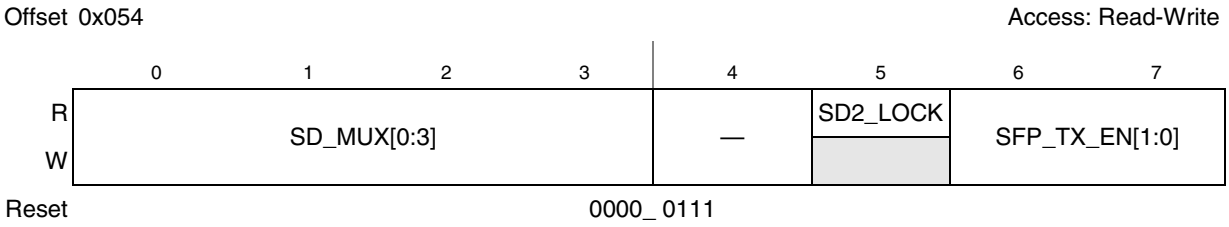


Figure 5-46. BRDCFG4 Register

Table 5-53. BRDCFG4 Register Field Descriptions

Bits	Name	Description
0-3	SD_MUX	Select Serdes lane's termination peripheral device (depends on SerDes protocol selection). ---------- A ----- B-----C------D-- 0011 - PCIe Slot CPRI2 CPRI1 0001 - PCIe Slot SGMII1 CPRI1 0000 - PCIe Slot SGMII1 SGMII2 (default) 0111 - PEX SGMII2 CPRI2 CPRI1 0101 - PEX SGMII2 SGMII1 CPRI1 1111 - SGMII1 SGMII2 CPRI2 CPRI1 All other values are reserved.
4	—	Reserved.
5	SD2_LOCK	SD2_REFCLK_LOCK status reflection. 0 - SD2 REFCLK PLL unlocked 1 - SD2 REFCLK PLL locked QIXIS SD2_REFCLK_LOCK input sampled
6-7	SFP_TX_EN	SFP optical transmitter control Controls SFP_TX_EN[0:1] FPGA output signals 0 - disable 1 - enable (default)

5.8.6 Board Configuration Register 5 (BRDCFG5)

The BRDCFG5 register controls signals multiplexing for USB based peripherals on the target system.

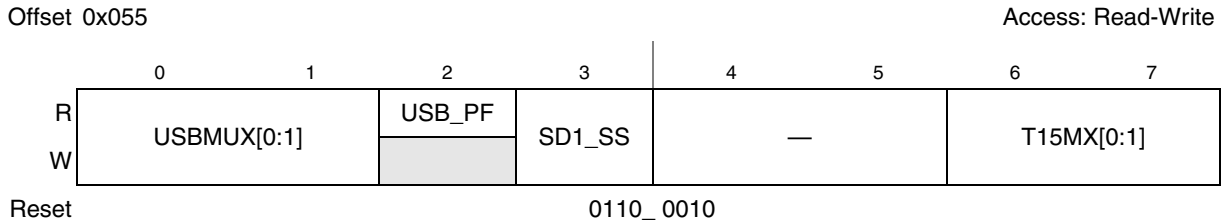


Figure 5-47. BRDCFG5 Register

Table 5-54. BRDCFG5 Register Field Descriptions

Bits	Name	Description
0-1	USBMUX	USB MUX Control: 00 Disconnected. 01 DUT ULPI is routed to USB PHY (default) 10 GPIO[53,57,62-63,69-72, TRIG_IN, and TIMER1 is routed to QIXIS 11 DUT I2C, UART2, GPIO[0:3] ().
2	USB_PF	USB Power Fault: 0 USB Power Fault 1 Normal operation. Sampled QIXIS USB_PWR_FAULTn input
3	SD1_SS	SD1_SS_MODE_B. 0 SD1 PLL CLK output is Not Spread Spectrum mode (default) 1 SD1 PLL CLK output is in a Spread Spectrum mode
4-5	—	Reserved.
6-7	T15MX	IEEE-TSEC_1588_ CLK_IN Pin Multiplexing Control: 00 CON_XCVR_REF clock from one of on board RF cards. 01 SYS CLOCK. 10 125Mhz from OCS or from 1588 Card if this card is plugged (default). 11 CON_XCVR_REF. QIXIS controls FPGA_1588_CLK_SEL[0:1] signals

5.8.7 Board Configuration Register 6 (BRDCFG6)

The BRDCFG6 registers are used to control the DUT Serial IF - UART demultiplexers.

Offset 0x056 Access: Read-Write

	0	1	2	3	4	5	6	7
R W	OCM_MUX	GPS_MUX	UART_MUX	—			ANT12CKM[0:1]	
Reset	0000_0000.							

Figure 5-48. BRDCFG6 Register

Table 5-55. BRDCFG6 Register Field Descriptions

Bits	Name	Description
0	OCM_MUX	UART1 MUX Control: 0 UART1 link routed to COM2 (default) 1 OCM IUART link routed to COM2. Sampled DCMSHELL on board SW (SW6.6). Control QIXIS CFG_OCM_UART output
1	GPS_MUX	UART3 GPS_MUX Control: 0 UART3 link is routed to J24 header (default). 1 UART3 link is routed to U136 GPS Module. Sampled DCMSHELL on board SW (SW6.6). Control QIXIS CFG_GPS_UART output

Table 5-55. BRDCFG6 Register Field Descriptions (continued)

Bits	Name	Description
2	UART_MUX	UART0 MUX Control: 0 UART0 link routed to COM1header. (default) 1 DUT GPIO[42:46],GPIO[56:57]. Control QIXIS UART_MUX_SEL_L output
3-5	—	Reserved.
6-7	ANT12CKM	DUT input ANT1 and ANT2 REF_CLK selection: 00 CON1_XCVR_REF is used as source for the DUT ANT1 ref clock (default) CON2_XCVR_REF is used as source for the DUT ANT2 ref clock (default) 10 CON3_XCVR_REF is used as source for the DUT ANT1 ref clock CON2_XCVR_REF is used as source for the DUT ANT2 ref clock 01 CON1_XCVR_REF is used as source for the DUT ANT1 ref clock CON3_XCVR_REF is used as source for the DUT ANT2 ref clock (J52.1-3 is closed) 01 CON3_XCVR_REF is used as source for the DUT ANT1 ref clock CON4_XCVR_REF is used as source for the DUT ANT2 ref clock (J52.2-3 is closed) QIXIS controls VCVR_REF_SEL[0:1] signals

5.8.8 Board Configuration Register 7 (BRDCFG7)

The BRDCFG7 registers are used to reflect on board SFP devices status.

Offset 0x057 Access: Read-Write

	0	1	2	3	4	5	6	7
R	SFPTX1	SFPTX2	RXLOS1	RXLOS2	I2C1ISO	—		
W								
Reset					0			

Figure 5-49. BRDCFG7 Register

Table 5-56. BRDCFG7 Register Field Descriptions

Bits	Name	Description
0	SFPTX1	SFP1 optical transceiver TX fault flag: 1 TX Fault 0 Normal operation Sampled SFP1_TX_FAULT inputs
1	SFPTX2	SFP2 optical transceiver TX fault flag: 1 TX Fault 0 Normal operation Sampled SFP2_TX_FAULT inputs
2	RXLOS1	SFP optical transceiver RX loss flag: 1 RX loss 0 Normal operation Sampled SFP_RX_LOS1 inputs

Table 5-56. BRDCFG7 Register Field Descriptions (continued)

Bits	Name	Description
3	RXLOS2	SFP optical transceiver RX loss flag: 1 RX loss 0 Normal operation Sampled SFP_RX_LOS2 inputs
4	I2C1ISO	ISO_I2C1 to DUT I2C1 bus switch control (sampled from SW_CFG_I2C1_ISO_EN): 0 Switch disconnected (default) 1 QIXIS and GSM I2C EEPROM are connected to DUT I2C1 bus controls QIXIS CFG_I2CSEL output
5-7	—	Reserved.

5.8.9 Board Configuration Register 8 (BRDCFG8)

The BRDCFG8 registers are used to reflect on manual CFG spare switch slots status.

Offset 0x058 Access: Read-Write

	0	1	2	3	4	5	6	7
R	JTAG_EE		SW_DBG		CKSTP0_O	CKSTP0_O	NU	IRQ_B
W								
Reset					0000			

Figure 5-50. BRDCFG8 Register

Table 5-57. BRDCFG8 Register Field Descriptions

Bits	Name	Description
0-1	JTAG_EE	PSC9132 JTAG Selection Read on board JTAG path SW selector Sampled SW_CFG_JTAG_MODE[0:1]
2-3	SW_DBG	SW_DBG[0:1] manual switch slots: 0 Off 1 On Switches mapping: SW_DBG[0:7] = {SW3.[7:8], SW2.[6:8], SW7.8, SW8.8}
4	CKSTP0_O	Checkstop out - Core 0 status 0 Negated 1 Asserted CKSTP0_OUT_B signal reflection
5	CKSTP1_O	Checkstop out - Core1. 0 Negated 1 Asserted CKSTP1_OUT_B signal reflection

Table 5-57. BRDCFG8 Register Field Descriptions (continued)

Bits	Name	Description
5	NU	Reserved
7	IRQ_B	DUT Interrupt request output - IRQ_OUT_B: 1 Acknowledge of DUT IRQn event 0 Nothing

5.8.10 Board Configuration Register 9 (BRDCFG9)

The BRDCFG9 registers controls some aspects of the target system power supply operation.

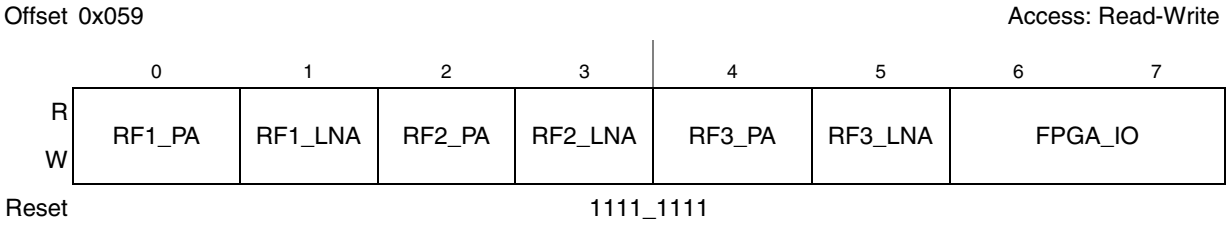


Figure 5-51. BRDCFG9 Register

Table 5-58. BRDCFG9 Register Field Descriptions

Bits	Name	Description
0	RF1_PA	RF1_CTRL_PA 1 PA regulator is on - high band enable (default) 0 PA regulator is off - low band enable
1	RF1_LNA	RF1_CTRL_LNA 1 LNA regulator is on - high band enable (default) 0 LNA regulator is off - low band enable
2	RF2_PA	RF2_CTRL_PA 1 PA regulator is on - high band enable (default) 0 PA regulator is off - low band enable
3	RF2_LNA	RF2_CTRL_LNA 1 LNA regulator is on - high band enable (default) 0 LNA regulator is off - low band enable
4	RF3_PA	RF3_CTRL_PA 1 PA regulator is on - high band enable (default) 0 PA regulator is off - low band enable
5	RF3_LNA	RF3_CTRL_LNA 1 LNA regulator is on - high band enable (default) 0 LNA regulator is off - low band enable
6-7	FPGA_IO	FPGA_GPIO[6:7] - auxiliary FPGA IO (header J17 pins 1 and 2 respective FPGA_GPIO[6] connected to the CP-RCLK (for current version)

5.8.11 Board Configuration Register 10 (BRDCFG10)

The BRDCFG10 registers are used to control and reflect on DUT external interrupts signals status.

Offset 0x05A Access: Read-Write

	0	1	2	3	4	5	6	7
R W	IRQ0	IRQ1	IRQ2	IRQ3	IRQ4	IRQ5	IRQ6	IRQ7-
Reset	0000_0000							

Figure 5-52. BRDCFG10 Register

Table 5-59. BRDCFG10 Register Field Descriptions

Bits	Name	Description
0	IRQ0	DUT external interrupt signal input: 0 Disabled (default) 1 Enabled Controls FPGA IRQ0 output
1	IRQ1	DUT external interrupt signal input: 0 Disabled (default) 1 Enabled Controls FPGA IRQ1 output
2	IRQ2	DUT external interrupt signal input: 0 Disabled (default) 1 Enabled Controls FPGA IRQ2 output
3	IRQ3	DUT external interrupt signal input: 0 Disabled (default) 1 Enabled Controls FPGA IRQ3 output
4	IRQ4	DUT external interrupt signal input: 0 Disabled (default) 1 Enabled Controls FPGA IRQ4 output
5	IRQ5	DUT external interrupt signal input: 0 Disabled (default) 1 Enabled Controls FPGA IRQ5 output
6	IRQ6	DUT external interrupt signal input: 0 Disabled (default) 1 Enabled Controls FPGA IRQ6 output
7	IRQ7	DUT external interrupt signal input: 0 Disabled (default) 1 Enabled Controls FPGA IRQ7 output

5.8.12 Board Configuration Register 11 (BRDCFG11)

The BRDCFG11 registers are used to control and reflect on DUT misc signals status.

Offset 0x05B Access: Read-Write

	0	1	2	3	4	5	6	7
R	MCP0	MCP1	CKSTP0	CKSTP1	TRIG_OUT	TRIG_IN	SYS_DMA _REQ/	SYS_DMA _DONE
W								
Reset	0	0	0	0	0	0	0	0

Figure 5-53. BRDCFG11 Register

Table 5-60. BRDCFG11 Register Field Descriptions

Bits	Name	Description
0	MCP0	Machine check exception - MCP_IN_B0 DUT output: 0 Disabled (default) 1 Enabled Controls FPGA MCP0_B output
1	MCP1	Machine check exception - MCP_IN_B1 DUT output: 0 Disabled (default) 1 Enabled Controls FPGA MCP1_B output
2	CKSTP0	Machine check to the e500 core 0 in response to checkstop from the e500 core 1. 0 Deassert (default) 1 Assert Controls FPGA CKSTOP0_IN_B output (and with COP_CKSTOP_IN signal)
3	CKSTP1	Machine check to the e500 core 1in response to checkstop from the e500 core 0. 0 Deassert. (default) 1 Assert Controls FPGA CKSTOP0_IN_B output
4	TRIG_OUT	Trigger out: 0 Nothing 1 Triggering of event was done
5	TRIG_IN	Trigger in. Can be used to trigger the watchpoint and trace buffers. Note this is an active-high (rising edge triggered) signal 0 -> 1 Initiate event 0 Nothing
6	SYS_DMA_REQ	Assert DUT SCAN_MODE input: 0 Deassert (default) 1 Assert ()
7	SYS_DMA_DONE	Indicate SYS DMA transfer: 0 Not action. (default) 1 DMA transfer is Done

5.8.13 Board Configuration Register 12 (BRDCFG12)

The BRDCFG12 registers are used to control and reflect on DUT misc signals status.

Offset 0x05C Access: Read-Write

	0	1	2	3	4	5	6	7
R	DBGREQ	DBGACK	UDE_B0	UDE_B1	SCAN	DMADACK	DMAREQ	DMADONE
W								
Reset		0		0	0	0	0	0

Figure 5-54. BRDCFG12 Register

Table 5-61. BRDCFG12 Register Field Descriptions

Bits	Name	Description
0	DBGREQ	Allow the DSP core to proceed to debug state after assertion of HReset: 0 Disabled (default) 1 Enabled Controls FPGA_DSP_DEBUG_REQ output
1	DBGACK	Status the DSP core in debug state after assertion of HReset: 0 Free running (default) 1 In debug state. Sampled FPGA_DSP_DEBUG_ACK inputs
2	UDE_B0	Assert DUT UDE_B0 input: 0 Deassert (default). 1 Assert
3	UDE_B1	Asset DUT UDE_B0 input: 0 Deassert. (default) 1 Assert
4	SCAN	DUT SCAN_MODE Factory Test. Refer to the chip Integrated Host Processor Hardware Specifications for proper treatmentAssert input: 0 Deassert (default) 1 Assert ()
5	DMADACK	Indicate DMADACK: 0 Nothing 1 DMA acknowledge
6	DMAREQ	Assert DMA_REQ DUT input: 0 Complete 1 Initiate DMA
7	DMADONE	Indicate DMA transfer: 0Not action. (default) 1DMA transfer is Done

5.9 DUT Configuration Registers

This block of 16 registers control the configuration of the DUT (Device Under Test). DUTCFG registers, unlike BRDCFG registers, are not always static. They are driven low only during the reset configuration sampling interval (HRESET_B assertion), and remain tri-stated thereafter; however, there are occasionally some DUT configuration signals that are static. For example TEST_SEL/TEST_SEL_B.

With the advent of the RCW, there are fewer pin-sampled configuration controls, and they are much more cross-platform compatible, so DUTCFG registers are mapped into a superset of available configuration controls. To facilitate software reuse, the first four DUTCFG registers have fixed bit assignments, of a superset across a wide variety of processors. The remainder are assigned as needed for each device. As with BRDCFG, see the relevant board/DUT documentation for details.

Table 5-62 shows the allocation of DUTCFG registers into fixed and floating varieties, with some processor-specific examples for the floating bits.

Register	bits	Fixed?	Description
DUTCFG0	0-7	Yes	Reserved.
DUTCFG1	0-1	Yes	DRAM type.
	2	Yes	IFC ECC enable/disable
	7	Yes	RCW extended mapping (if applicable).
DUTCFG2	0-1	Yes	Testport Modes (TPORT_MX + TPORT_DIS)
	4-6	Yes	SVR[2:0]
	7	Yes	TEST_SEL / TEST_SEL_B
DUTCFG[3]	0-7	Yes	VSEL
DUTCFG[4]	0-7	Yes	SerDes Ports
DUTCFG[5]	0-7	Yes	Reserved
DUTCFG6	0	No	PSC9132: CFG SYS PLL
DUTCFG7	1	No	PSC9132: CFG CORE PLL
DUTCFG8	2	No	PSC9132: CFG DSP PLL
DUTCFG9	3	No	PSC9132: CFG DDR PLL
DUTCFG10	4	No	PSC9132: CFG IFC
DUTCFG11	5	No	PSC9132: CFG BOOT
DUTCFG12	6	No	PSC9132: CFG Customer Unvisible
DUTCFG13	7	No	PSC9132: CFG ENG USE
DUTCFG14	0-7	No	P1010 CFG (Interposer)
DUTCFG15	7	No	Reserved

Table 5-62. DUTCFG*n* Customary Bit Assignments

5.9.1 DUT Configuration Register 0 (DUTCFG0)

The DUTCFG0 register is used to control fixed DUT configurations, in particular the RCW location setting (cfg_rcw_src).

Offset 0x060 Access: Read-Write

0	1	2	3	4	5	6	7

Figure 5-55. DUTCFG0 Register

5.9.2 DUT Configuration Register 1 (DUTCFG1)

The DUTCFG1 register is used to control additional fixed DUT configurations, including DDR DRAM type, ECC enable. For PSC9132, only EECC is valid.

Offset 0x061 Access: Read-Write

	0	1	2	3	4	5	6	7
R W	D1_DDR_TYPE	D2_DDR_TYPE	EECC		—			
Reset	Sampled from switches.							

Table 5-64. DUTCFG1 Register Field Descriptions

Bits	Name	Description
0	D1_DDR_TYPE	Controls cfg_d1_dram_type.
1	D2_DDR_TYPE	Controls cfg_d2_dram_type.
2-3	EECC	Controls cfg_ifc_ecc[0..1].
4-7	—	Reserved.

5.9.3 DUT Configuration Register 2 (DUTCFG2)

The DUTCFG2 register is used to control fixed DUT configurations, particularly testport mode and SVR variations.

Offset 0x062 Access: Read-Write

	0	1	2	3	4	5	6	7
R	TP_MX	TP_DIS	—			SVR		TEST
W								
Reset	Sampled from switches.							

Figure 5-56. DUTCFG2 Register

Table 5-65. DUTCFG2 Register Field Descriptions

Bits	Name	Description
0	TP_MX	Controls *cfg_test_port_mux_sel. (low by default - not selected)*
1	TP_DIS	Controls cfg_test_port_dis*(high by default - disable).*
2-4	—	Reserved.
4-5	SVR	Controls cfg_svr[1..0].
5-6	SVR	Controls cfg_svr[1..0].
7	TEST	Controls *TEST_SEL_B. (high by default - not selected)* Note that this one special DUT configuration signal is static.

5.9.4 DUT Configuration Register 3 (DUTCFG3)

The DUTCFG3 register is used to control fixed DUT configurations, particularly VSEL mode of chip.

Offset 0x063 Access: Read-Write

	0	1	2	3	4	5	6	7
R	—	SIM_VSEL	BVDD_VSEL0	BVDD_VSEL1	CVDD_VSEL	LVDD_VSEL	X1VDD_VSEL	X2VDD_VSEL
W								
Reset	Sampled from switches.							

Figure 5-57. DUTCFG3 Register

Table 5-66. DUTCFG3 Register Field Descriptions

Bits	Name	Description
0	—	Reserved.
1	SIM_VSEL	**SIM Card Voltage Selection** (SIM_VSEL) 0 - 1.8V 1 - 3.3V
2-3	BVDD_VSEL	**DUT BVDD Voltage Selection** (BVDD_SEL[0:1]) 2'b00 - 3.3V 2'b01 - 2.5V 2'b10 - 1.8V 2'b11 - Reserved
4	CVDD_VSEL	**DUT CVDD Voltage Selection** (CVDD_SEL) 0 - 3.3V 1 - 1.8V
5	LVDD_VSEL	**DUT LVDD Voltage Selection** (LVDD_SEL) 0 - 3.3V 1 - 2.5V

Table 5-66. DUTCFG3 Register Field Descriptions (continued)

Bits	Name	Description
6	X1VDD_VSEL	**DUT XVDD Voltage Selection** (XVDD_SEL) 0 - X1VDD = X2VDD = 3.3V 1 - X1VDD = X2VDD = 1.8V
7	X2VDD_VSEL	**DUT XVDD Voltage Selection** (XVDD_SEL) 0 - X1VDD = X2VDD = 3.3V 1 - X1VDD = X2VDD = 1.8V

5.9.5 DUT Configuration Register 4 (DUTCFG4)

The DUTCFG4 registers are used to control DUT SerDesconfiguration pins.

Offset DUTCFG4: 0x064 Access: Read-Write

	0	1	2	3	4	5	6	7
R	SRDS_IO_PORTS[0:6]							HOST_AGT
W								

Reset Varies; sampled from switches or defaults to a DUT-internal RCW

Figure 5-58. DUTCFG3-DUTCFG4 Register

Table 5-67. DUTCFG4 Register Field Descriptions

Bits	Name	Description
0-6	SRDS_IO_PORTS	Controls *cfg_srds_io_ports[0..6]*. For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM)*.
7	HOST_AGT	Controls *cfg_host_agt*. For details, see the Chapter 4 “Reset and Initialization” in BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).

5.9.6 DUT Configuration Register 5 (DUTCFG5)

The DUTCFG5 registers are used to control fixed DUT configuration pins. They are currently not assigned.

Offset DUTCFG5: 0x065 Access: Read-Write

	0	1	2	3	4	5	6	7
R	—							
W								

Reset Varies; sampled from switches or defaults to a DUT-internal RCW

Figure 5-59. DUTCFG5 Register

Table 5-68. DUTCFG5 Register Field Descriptions

Bits	Name	Description
0-7	—	Reserved.

5.9.7 DUT Configuration Register 6 (DUTCFG6)

The DUTCFG6 is used to sample device-specific configuration modes, particularly DUT SYS_PLL

Offset 0x066 Access: Read-Write

	0	1	2	3	4	5	6	7
R W	SYS_PLL[0:2]			SYS_SPEED	PLAT_SPEED	SRDS_REFCLK	—	
Reset	Sampled from switches.							

Figure 5-60. DUTCFG6 Register

Table 5-69. DUTCFG6 Register Field Descriptions

Bits	Name	Description
0-2	SYS_PLL	**CCB Clock PLL Ratio (**cfg_sys_pll[0:2]**)** 3'b000 - 4:1 3'b001 - 5:2 3'b010 - 6:3 3'b010... - 3'b111 - Reserved
3	SYS_SPEED	**System Speed (**cfg_sys_speed**)** 0 -> 33 MHz > SYSCLK frequency < 65 MHz 1 -> 66 MHz > SYSCLK frequency < 133 MHz
4	PLAT_SPEED	Platform Speed (cfg_plat_speed) 0 -> 320 MHz > CCB Clock frequency < 167 MHz 1 -> 320 MHz > CCB Clock frequency < 601 MHz **SerDes Reference Clock Configuration (**cfg_srds_
5	SRDS_REFCLK	**SerDes Reference Clock Configuration (**cfg_srds_refclk**)** 0 -125 MHz reference clock frequency. 1 -100 MHz reference clock frequency.
6-7	—	Reserved.

5.9.8 DUT Configuration Register 7(DUTCFG7)

The DUTCFG7 is used to sample device-specific configuration modes, particularly DUT SYS_PLL.

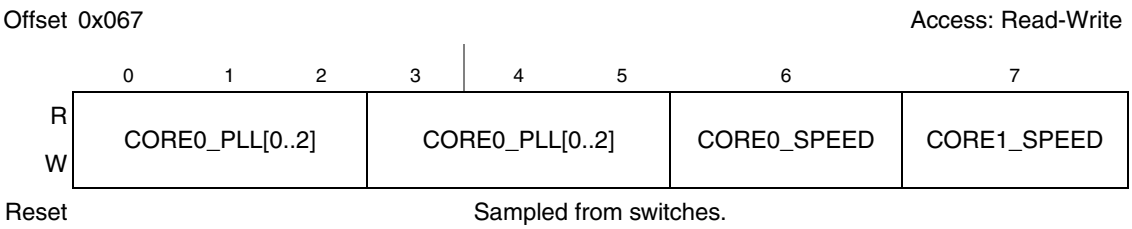


Figure 5-61. DUTCFG7 Register

Table 5-70. DUTCFG7 Register Field Descriptions

Bits	Name	Description
0-2	CORE0_PLL	**e500 Core 0: CCB Clock Ratio** (cfg_core0_pll[0:2]) 3'b010 1:1 3'b011 3:2 3'b000 - 4:1 3'b001 - 5:2 3'b100 - 2:1 3'b110 - 3:1 3'b000... 3'b001,3'b111 - Reserved.
3-5	CORE1_PLL	**e500 Core 1: CCB Clock Ratio** (cfg_core1_pll[0:2]) 3'b100 - 2:1
6	CORE0_SPEED	Controls *cfg_core0_speed.* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM)*.
7	CORE1_SPEED	Controls*cfg_core1_speed.* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM)*.

5.9.9 DUT Configuration Register 8(DUTCFG8)

The DUTCFG8 is used to sample device-specific configuration modes, particularly DSP PLL.

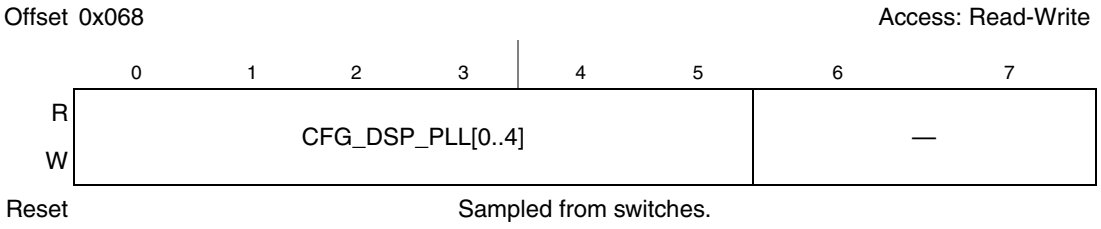


Figure 5-62. DUTCFG8 Register

Table 5-71. DUTCFG8 Register Field Descriptions

Bits	Name	Description
0-5	CFG_DSP_PLL	Controls *cfg_dsp_pll[0..4].* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
6-7	—	Reserved.

5.9.10 DUT Configuration Register 9(DUTCFG9)

The DUTCFG9 is used to sample device-specific configuration modes, particularly DDR PLL.

Offset 0x069 Access: Read-Write

	0	1	2	3	4	5	6	7
R W	D1_DDR_PLL[0..1]		D1_DDR_SPEED[0..1]		—		D1_DDR_HALF_FULL_MODE	D1_DDR_HALF_FULL_MODE
Reset	Sampled from switches.							

Figure 5-63. DUTCFG9 Register

Table 5-72. DUTCFG9 Register Field Descriptions

Bits	Name	Description
0-1	D1_DDR_PLL	Controls *cfg_d1_ddr_pll0..1].* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
2-3	D1_DDR_SPEED	Controls *cfg_d1_ddr_speed[0..1].* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
4-5	—	Reserved.
6	D1_DDR_HALF_FULL_MODE	Controls *cfg_d1_ddr_half_full_mode.* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
7	D2_DDR_HALF_FULL_MODE	Controls *cfg_d1_ddr_half_full_mode.* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*

5.9.11 DUT Configuration Register 10(DUTCFG10)

The DUTCFG10 is used to sample device-specific configuration modes, particularly DUT IFC configuration.

Offset 0x06A Access: Read-Write

	0	1	2	3	4	5	6	7
R W	CFG_IFC_PB[0..2]			—	IFC_ADM_MODE	IFC_FLASH_MODE		—
Reset	Sampled from switches.							

Figure 5-64. DUTCFG10 Register

Table 5-73. DUTCFG10 Register Field Descriptions.

Bits	Name	Description
0-2	CFG_IFC_PB	Controls *cfg_ifc_pb[0..2].* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
3-4	—	Reserved.
5	IFC_ADM_MODE	Controls *cfg_ifc_adm_mode.* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
6	IFC_FLASH_MODE	Controls *cfg_ifc_flash_mode.* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
7	—	Reserved

5.9.12 DUT Configuration Register 11(DUTCFG11)

The DUTCFG11 is used to sample device-specific configuration modes, particularly DUT BOOT configuration.

Offset 0x06B Access: Read-Write

	0	1	2	3	4	5	6	7
R W	CFG_ROM_LOC[0..3]				CPU0_BOOT	BOOT_SEQ[0:1]		—
Reset	Sampled from switches.							

Figure 5-65. DUTCFG11 Register

Table 5-74. DUTCFG11 Register Field Descriptions

Bits	Name	Description
0-3	CFG_ROM_LOC	Controls *cfg_rom_loc[0..3].* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
4	CPU0_BOOT	Controls *cpu0_boot.* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
5-6	BOOT_SEQ	Controls *cfg_boot_seq[0:1].* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
7	—	Reserved

5.9.13 DUT Configuration Register 12 (DUTCFG12)

The DUTCFG12 register is used to control the customer invisible DUT configuration signals.

Offset DUTCFG12: 0x06C — Access: Read-Write

	0	1	2	3	4	5	6	7
R W	FUSE_RD_EN	BIST	PPC_DROWSY_EN	DSP_DROWSY_EN	SB_DIS	SRDS_PLL_TIMEOUT_EN	D1_DDR_PLL_BACKUP	D1_DDR_PLL_BACKUP
Reset	Sampled from switches.							

Figure 5-66. DUTCFG12 Register

Table 5-75. DUTCFG12 Register Field Descriptions

Bits	Name	Description
0	FUSE_RD_EN	Controls *cfg_fuse_rd_en (low by default - disabled).* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
1	BIST	Controls *cfg_por_bist (low by default - disabled).* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
2	PPC_DROWSY_EN	Controls *cfg_ppc_drowsy_en (low by default - disabled).* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
3	DSP_DROWSY_EN	Controls *cfg_dsp_drowsy_en (low by default - disabled).* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*

Table 5-75. DUTCFG12 Register Field Descriptions (continued)

Bits	Name	Description
4	SB_DIS	Controls *cfg_sb_dis (high by default - disabled).* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
5	SRDS_PLL_TIMEOUT_EN	Controls *cfg_srds_pll_timeout_en (high by default - enabled).* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
6	D1_DDR_PLL_BACKUP	Controls *cfg_d1_ddr_pll_backup (low by default - disabled).* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
7	D1_DDR_PLL_BACKUP	Controls *cfg_d2_ddr_pll_backup.(low by default - disabled)* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*

5.9.14 DUT Configuration Register 13 (DUTCFG13)

The DUTCFG13 register is used to control the CFG_ENGUSE[0:1] configuration signals.

Offset DUTCFG13: 0x06D — Access: Read-Write

	0	1	2	3	4	5	6	7
R W	ENG_USE		60X		—			
Reset	Sampled from switches.							

Figure 5-67. DUTCFG13 Register

Table 5-76. DUTCFG13 Register Field Descriptions

Bits	Name	Description
0-1	ENG_USE	Controls *cfg_eng_use[0:1]. (by default = 2’b00)* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
2	60X	Controls *cfg_60X. (low by default)* For details, see the Chapter 4 “Reset and Initialization” in *BSC9132 QorIQ Qonverge Multicore Baseband Processor Reference Manual (document BSC9132RM).*
3-7	—	Reserved.

5.9.15 DUT Configuration Register 14-15 (DUTCFG14-15)

The DUTCFG14-DUTCFG15 registers are used to control the general-purpose GPCFG signals. These configurations are used exclusively by end customers for their own purposes.

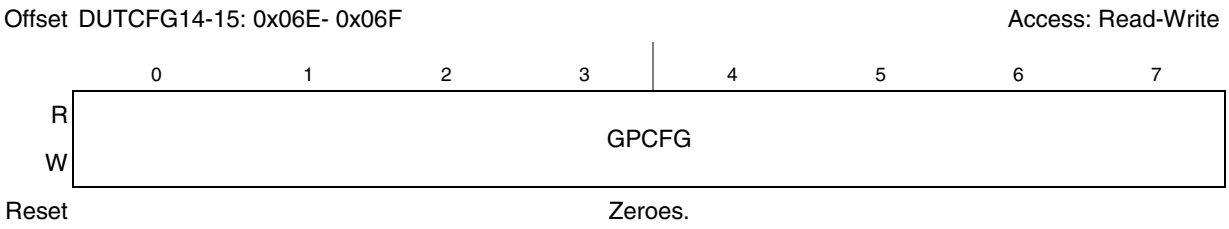


Figure 5-68. DUTCFG14-15 Register

Table 5-77. DUTCFG14-5 Register Field Descriptions

Bits	Name	Description
0-7	—	DUTCFG14 -15: reserved

5.10 RCW Access Registers

The Reset Control World (RCW) access registers provide read/write capabilities to the 512B RCW SRAM in the **QIXIS**. This access is primarily for remote I2C-based controllers, as the DUT has direct access through the IFC. However, it works for the DUT as well.

5.10.1 RCW SRAM Address Registers (RCW_ADDR*n*)

The RCW_ADDR*n* registers store the address index of the RCW SRAM array; RCW_ADDR0 represents the MSB and RCW_ADDR1 represents the LSB. The 16-bit address is fully readable/writable. However, the actual size of the RCW SRAM may be significantly smaller. See the target system documentation for details.

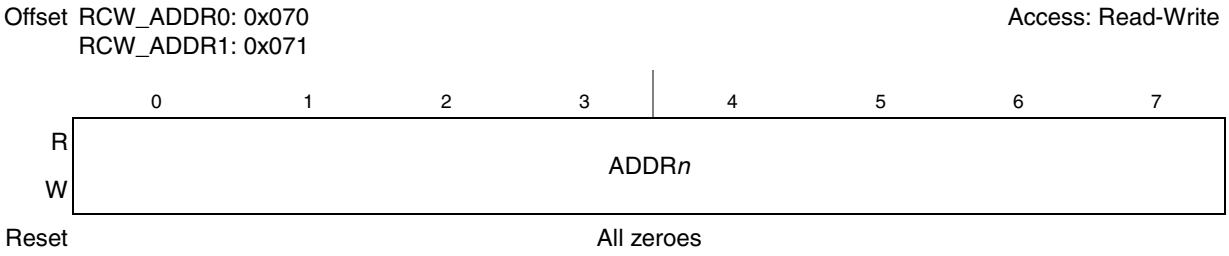


Figure 5-69. RCW_ADDR*n* Registers

Table 5-78. RCW_ADDR*n* Registers Field Description

Bits	Name	Description
0-7	ADDR*n*	MSB (ADDR0) or LSB (ADDR1) of the RCW address.

5.10.2 RCW SRAM Data Register (RCW_DATA)

The RCW_DATA register is used to read from or write to the RCW SRAM array, at the address specified by the RCW_ADDR*n* registers. As SRAM arrays are not preset to 0 upon powerup, this register will present random values unless the SRAM is preset by software.

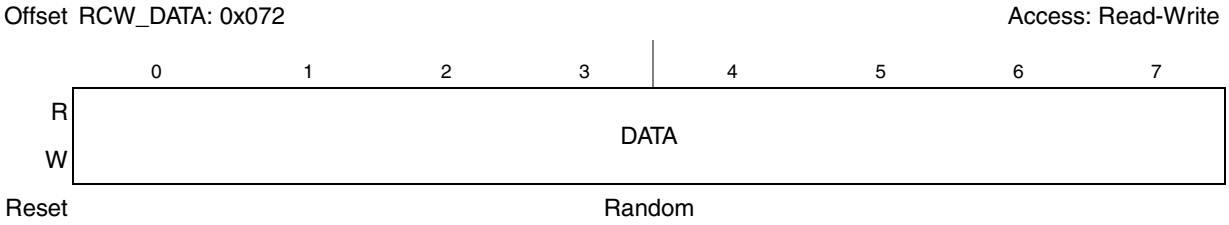


Figure 5-70. RCW_DATA Register

Table 5-79. RCW_DATA Register Field Description

Bits	Name	Description
0-7	DATA	Contents of SRAM array indexed by RCW_ADDR[0:1].

5.11 GPIO Control Registers

The GPIO registers are used to provide direct control of GPIO signals; up to 32 pins can be directly controller and monitored. In all cases, the behaviour of a GPIO pin is controlled by a bit in a corresponding GPIO_DIR*n* register, and a corresponding bit in a GPIO_IO register.

Table 5-80. GPIO Behaviour

GPIO Settings		Description
GPIO_DIR*n.b*	GPIO_IO*n.b*	
0	0	GPIO[n*8+b] is an input, currently at ‘0’.
0	1	GPIO[n*8+b] is an input, currently at ‘1’.
1	0	GPIO[n*8+b] is an output, currently driving ‘0’.
1	1	GPIO[n*8+b] is an output, currently driving ‘1.

5.11.1 GPIO I/O Signal Registers (GPIO_IO*n*)

GPIO_IO*n* registers show the current value of GPIO signals (for those programmed as an input) or the current driven signal level (for those programmed as an output).

Note that writes to this register behave according to the programming of the corresponding bits in a GPIO_DIR*n* register. If a bit is programmed as an input, the bit in the GPIO_IO*n* register cannot be changed.

Offset GPIO_IO0: 0x080, GPIO_IO1: 0x081, GPIO_IO2: 0x082, GPIO_IO3: 0x083 — Access: Read/Write

	0	1	2	3	4	5	6	7
R W	GPIO0 GPIO53 GPIO72	GPIO1 GPO56 GPO73	GPIO5 GPIO57 GPIO80	GPO6 GPIO62 TIMER1/ GPIO2	GPIO42 GPIO63 NU/	GPO43 GPIO69 TIMER5	GPIO44 GPIO70 TIMER6/ GPIO23	GPO45 GPIO71 TIMER7/ GPIO24
Reset	Input reflection							

Figure 5-71. GPIO_IO*n* Registers

Table 5-81. GPIO_IO*n* Register Field Descriptions

Bits	Name	Description
0-7	GPIO*n*.b	Current value of corresponding GPIO pin, whether an input or a driven output.

5.11.2 GPIO Direction Registers (GPIO_DIR*n*)

GPIO_DIR*n* registers control the direction of GPIO signals; if a bit in a GPIO_DIR*n* is set to one, the signal is an output, set to the value in the GPIO_IO*n* register; otherwise it is an input whose value is shown in the GPIO_IO*n* register.

Offset GPIO_DIR0: 0x084, GPIO_DIR1: 0x085, GPIO_DIR2: 0x086, GPIO_DIR3: 0x087 — Access: Read/Write

	0	1	2	3	4	5	6	7
R W	DIR0 DIR8 DIR16 DIR24	DIR1 DIR9 DIR17 DIR25	DIR2 DIR10 DIR18 DIR26	DIR3 DIR11 DIR19 DIR27	DIR4 DIR12 DIR20 DIR28	DIR5 DIR13 DIR21 DIR29	DIR6 DIR14 DIR22 DIR30	DIR7 DIR15 DIR23 DIR31
Reset	All zeroes							

Figure 5-72. GPIO_DIR*n* Registers

Table 5-82. GPIO_DIR*n* Register Field Descriptions

Bits	Name	Description
0-7	DIR*n*.b	GPIO Direction: 0 The pin is an input pin. 1 The pin is an output pin.

5.12 Remote JTAG Access Registers

To perform a limited JTAG/COP status controlling and monitoring through the remote I2C traffic, a small block of registers provide the ability to synthesize JTAG traffic to the DUT. This is a very low-level function, and data is read/written at the bit and byte level.

Note that JTAG ports might be shared among COP headers and Aurora ports, so BRDCFG-specified multiplexor control might be required.

5.12.1 Remote JTAG Control Register (RJTAG_CTL)

The RJTAG_CTL register allows direct manipulation of JTAG output signals.

Offset 0x09C Access: Read/Write

	0	1	2	3	4	5	6	7
R W	TRST	—		AUTO	—	TCK	TMS	TDI
Reset	1000_0000							

Figure 5-73. RJTAG_CTL Register

Table 5-83. RJTAG_CTL Register Field Descriptions

Bits	Name	Description
0	TRST	JTAG TRST Signal: 0 The TRST_B signal is forced low. 1 The TRST_B signal is not affected. Other reset sources may affect TRST_B.
1-2	—	Reserved.
3	AUTO	JTAG Automatic TCK: 0 The TCK signal is driven one bit at a time. 1 The TCK signal will be driven high, then low 8 times on each write to the RJTAG_DATA register.
4	—	Reserved.
5	TCK	JTAG TCK Signal: 0 The TCK signal is not affected. 1 The TCK signal is forced high. NOTE: This bit must be 0 to use the AUTO feature.
6	TMS	JTAG TMS Signal: 0 The TMS signal is not affected. 1 The TMS signal is forced high.
7	TDI	JTAG TDI Signal: 0 The TDI signal is not affected. 1 The TDI signal is forced high. **Note:** This bit must be 0 to use the AUTO feature.

5.12.2 Remote JTAG Data Register (RJTAG_DATA)

The RJTAG_DATA register is used to send/receive data to the DUT JTAG interface, whether a bit at a time, or 8 bits at a time. Note that in byte mode, data is shifted in or out the LSB first.

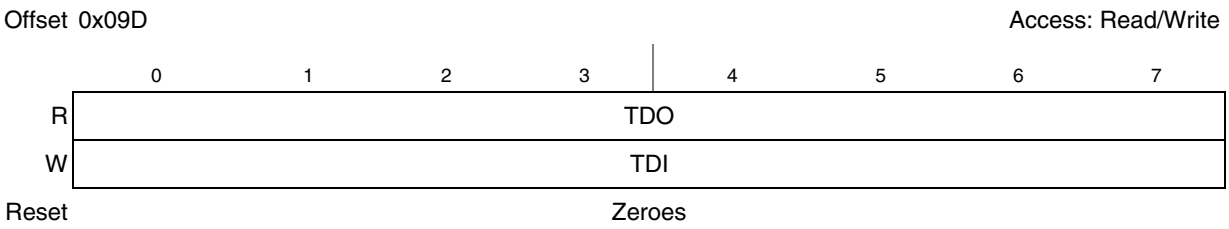


Figure 5-74. RJTAG_DATA Register

Table 5-84. RJTAG_DATA Register Field Descriptions

Bits	Name	Description
0-7	TDO	JTAG TDO: Either 8 bits of data sampled from the DUT TDO output (RJTAG_CTL[AUTO] = 1) or the live signal on the DUT TDO output ((RJTAG_CTL[AUTO] = 0).
0-7	TDI	JTAG TDI: Writes to this register trigger the AUTO clock mode (if enabled) and also provide the data to transmit to the DUT JTAG interface.

5.13 Auxiliary Registers

Auxiliary registers store data for software purposes. Other than initializing registers (but not SRAM) to 0 upon initial power-up, **QIXIS** does not alter the contents for any reason. Software may use these registers to store configuration data and other parameters across reset occurrences.

If more than 4 registers are needed, the AUX_ADDR and AUX_DATA provide indirect access to 256B of SRAM.

5.13.1 Auxiliary Registers (AUX1-AUX4)

The AUX*n* registers store software-dependant values. The AUX register is initialized to 0 when the system is powered-up. Once initialized, the register value is not changed by Qixis again.

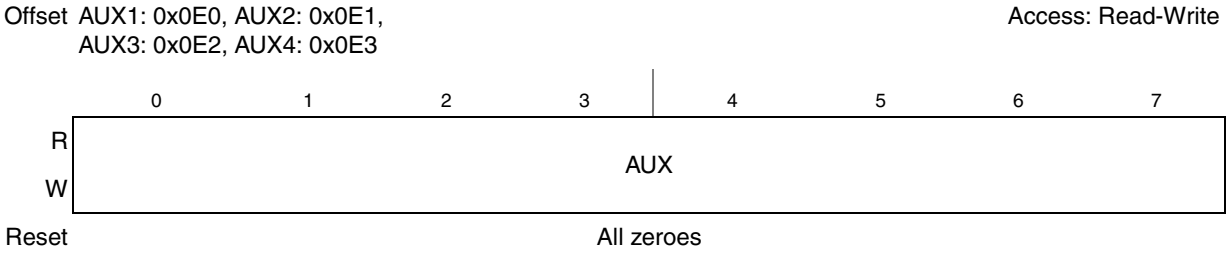


Figure 5-75. AUX*n* Registers

Table 5-85. AUX*n* Register Field Descriptions

Bits	Name	Description
0-7	AUX	User-supplied value.

5.13.2 Auxiliary SRAM Address Register (AUX_ADDR)

The AUX_ADDR register stores the address index of the auxiliary SRAM array. Similar to other AUX registers, AUX_ADDR register is initialized to 0 when the system powers up, and then the value is not updated by Qixis again.

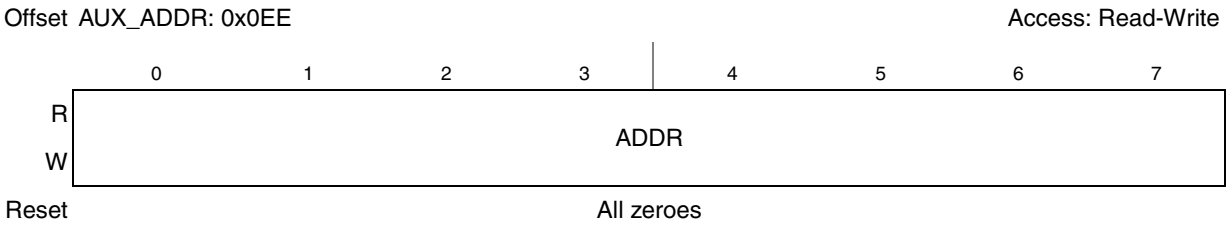


Figure 5-76. AUX_ADDR Register

Table 5-86. AUX_ADDR Register Field Description

Bits	Name	Description
0-7	ADDR	Index into 256B auxiliary SRAM array.

5.13.3 Auxiliary SRAM Data Register (AUX_DATA)

The AUX_DATA register is used to read from or write to the auxiliary SRAM array, at the address specified by the AUX_ADDR register. As SRAM arrays are not preset to 0 upon powerup, this register will present random values unless the SRAM is preset by software.

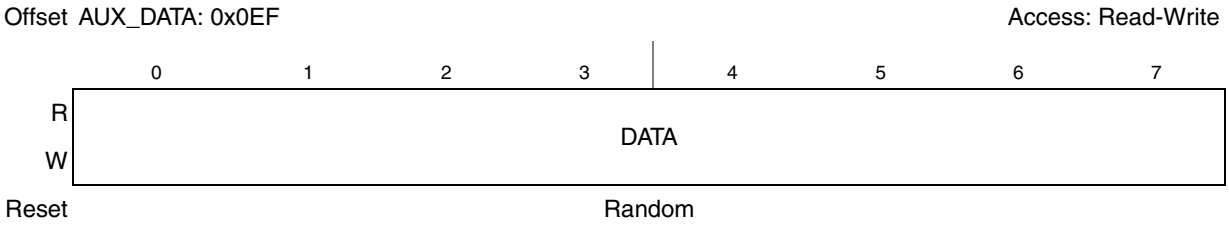


Figure 5-77. AUX_DATA Register

Table 5-87. AUX_DATA Register Field Description

Bits	Name	Description
0-7	DATA	Contents of SRAM array indexed by AUX_ADDR.

Appendix A
Revision History

A.1 Revision history

The table below provides the revision history for this document.

Table A-1 Revision history

Revision number	Date	Topic cross-reference	Change description
Rev. 0	05/2014		Initial public release.