**Lime Microsystems Limited**

Surrey Technology Centre

Occam Road

The Surrey Research Park

Guildford, Surrey GU2 7YG

United Kingdom

Tel: +44 (0) 1483 685 063

e-mail: [enquiries@limemicro.com](mailto:enquiries@limemicro.com)

**LimeSDR-QPCIe**

***- FPGA Gateware Description****-*

REVISION HISTORY

The following table shows the revision history of this document:

|  |  |  |
| --- | --- | --- |
| **Date** | **Version** | **Description of Revisions** |
| 26/11/2018 | 1.0 | Initial version |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Table of Contents**

[REVISION HISTORY 2](#_Toc519152725)

[1 Introduction 4](#_Toc519152726)

[2 FPGA gateware features 5](#_Toc519152727)

[3 Gateware description 6](#_Toc519152728)

[3.1 Main block diagram 6](#_Toc519152729)

[3.2 Clock network 8](#_Toc519152730)

[3.3 Softcore processor – nios\_cpu 9](#_Toc519152731)

[3.3.1 Registers of fpgacfg module 14](#_Toc519152732)

[3.3.2 Registers of pllcfg module 17](#_Toc519152733)

[3.3.3 Registers of tstcfg module 20](#_Toc519152734)

[3.3.4 Registers of periphcfg module 22](#_Toc519152735)

[3.4 PCIe interface – pcie\_top 27](#_Toc519152736)

[3.5 LMS7002 Receive and transmit interface – rxtx\_top 32](#_Toc519152737)

[3.5.1 Receive interface – rx\_path\_top 34](#_Toc519152738)

[3.5.2 Transmit interface – tx\_path\_top 35](#_Toc519152739)

[3.5.3 Waveform player – wfm\_player\_top **Error! Bookmark not defined.**](#_Toc519152740)

[3.6 General periphery – general\_periph\_top 44](#_Toc519152741)

[3.7 PLL module – pll\_top 47](#_Toc519152742)

[3.8 Board test module – tst\_top **Error! Bookmark not defined.**](#_Toc519152743)

[4 Examples 51](#_Toc519152744)

[4.1 Accessing FPGA registers 51](#_Toc519152745)

[4.2 Accessing LMS7002M registers 52](#_Toc519152746)

[4.3 Periphery control 53](#_Toc519152747)

[4.4 Configuring FPGA PLL module 53](#_Toc519152748)

[4.4.1 RX PLL module - rxpll\_top configuration (auto phase shift mode) 53](#_Toc519152749)

[4.4.2 TX PLL module - txpll\_top configuration (auto phase shift mode) 54](#_Toc519152750)

[4.5 Controlling TX and RX data stream 55](#_Toc519152751)

[4.6 Using WFM player 56](#_Toc519152752)

# Introduction

This document contains functional description of FPGA gateware project suited for LimeSDR-QPCIe board.

**FPGA project** - LimeSDR-QPCIe\_lms7\_trx project can be downloaded from GitHub repository <https://github.com/myriadrf/LimeSDR-QPCIe_GW>.

**Required hardware** – LimeSDR-QPCIe v1.2 board.

**Development software** – project is created with Intel Quartus prime, Version 18.0.0 Build 614 04/24/2018 SJ Lite Edition with Cyclone V GX device support. Mentioned software edition is free and can be downloaded from [(https://www.intel.com)](https://www.intel.com/). Although other Intel Quartus prime software versions supporting Cyclone V GX family might work as well but it is recommended to use same version as project was created.

# FPGA gateware features

Gateware contains following features:

* Interface to LMS7002 LimeLightTM digital IQ interface in TRXIQ double data rate mode;
* Real time data transfer between host and LMS7002 chip;
* PCIe interface for transferring data between host and FPGA;
* Interface to external ADC - ADS4246 and DAC - DAC5672.
* TX samples synchronization with RX samples time stamp;
* SPI connection between LMS7002 chip and other on-board devices;
* Reconfigurable PLL blocks for LMS7002 clocking;
* Internal SPI registers for FPGA control.

# Gateware description

This chapter describes main modules of LimeSDR-QPCIe\_lms7\_trx project.

## Main block diagram

Cyclone V FPGAprovides FIFO interface with PCIe. There are two endpoints (F2H\_C0 – FPGA to Host and H2F\_C0 – Host to FPGA) implemented for control data and six endpoints for stream data (H2F\_S0, H2F\_S1, H2F\_S2 – Host to FPGA and F2H\_S0, F2H\_S1, F2H\_S2 – FPGA to Host). Control endpoints are connected to NIOS II softcore processor which provides SPI and I2C communication interfaces for LMS7002M chip, TCXO DAC, ADF4002 phase detector, LM75 temperature sensor, Si5351C clock generator, ADS2426 Analog to Digital converter. NIOS also provides access to internal SPI configuration registers. Stream endpoints are dedicated for receiving and sending IQ data from/to LMS7002M and external DAC and ADC. **Figure 1** contains top block diagram with main modules. Description of main FPGA instances can be found in **Table 1**.



Figure 1 Top block diagram

Table 1 Description of main instances

| **Instance** | **Description** |
| --- | --- |
| nios\_cpu | NIOS II softcore processor with memory registers. Provides periphery control. See **3.3 Softcore processor – nios\_cpu.** |
| pcie\_top | Provides data transfer between external host and FPGA through PCIe interface See **3.4 PCIe interface – pcie\_top.** |
| rxtx\_top | Receive and transmit IQ data packets logic between FPGA and external LMS7002 transceiver or external DAC and ADC. See **3.5 Packet receive and transmit interface – rxtx\_top.** |
| LMS7002\_top | Module for receiving and transmitting IQ samples from/to LMS7002 DIQ bus **3.7 LMS7002 interface – lms7002\_top.** |
| adc\_top | Receive data from external ADC **3.8 External ADC – adc\_top.** |
| dac5672\_top | Transmit data to external DAC **3.9 External DAC – dac5672\_top.** |
| wfm\_player\_x2\_top | Waveform player **3.6 Waveform player – wfm\_player\_x2\_top.** |
| general\_periph\_top | Control module for onboard periphery such as LEDs, GPIO, FAN. See **3.10 General periphery – general\_periph\_to.** |
| pll\_top | Module provides required clocks for rxtx\_top module. See **3.11 PLL module – pll\_top.** |
| tst\_top | Board test logic to test external DDR2 memory and external clocks. See Error! Reference source not found.Error! Reference source not found.**.** |

## Clock network

**Figure 2** shows dataflow between main modules and clocking scheme. More details can be found in **Table 2**.



Figure 2 Gateware clock network

Table 2 Clock network description

| **Clock name** | **Frequency, MHz** |  | **Description** |
| --- | --- | --- | --- |
| LMS1\_MCLK1, LMS2\_MCLK1 | Configurable | | Sample clock from LMS7002M IC. Used as a reference clock for TXPLL. |
| LMS1\_MCLK2,  LMS2\_MCLK2 | Configurable | | Sample clock from LMS7002M IC. Used as a reference clock for RXPLL. |
| LMS1\_FCLK1,  LMS2\_FCLK1 | Configurable | | Sample clock, LMS7002M IC latches LMS\_DIQ1 bus signals using this clock. |
| LMS1\_FCLK2,  LMS2\_FCLK2, | Configurable | | Not used |
| lms1\_txpll\_c1,  lms2\_txpll\_c1, | Configurable | | FPGA launches LMS\_DIQ1 bus signals using this clock. Used for clocking FPGA TX modules. |
| lms1\_rxpll\_c1,  lms2\_rxpll\_c1, | Configurable | | FPGA latches LMS\_DIQ2 bus signals using this clock. Used for clocking FPGA RX modules. |
| CLK\_LMK\_FPGA\_IN | 30.72 | | Reference clock from LMK00105 clock buffer. |
| CLK125\_FPGA\_TOP | 125 | | External oscillator, used for DDR3\_TOP memory controller. |
| CLK125\_FPGA\_BOT | 125 | | External oscillator, used for DDR3\_BOT memory controller. |
| CLK100\_FPGA | 100 | | External oscillator, used for PCIe controller. |
| CLK125\_FPGA | 125 | | External oscillator, currently not used. |
| SI\_CLK0 | 27 | | Connected only to tst\_top module |
| SI\_CLK1 | 27 | |
| SI\_CLK6 | 27 | |
| SI\_CLK7 | 27 | |

## Softcore processor – nios\_cpu

**Figure 3** shows block diagram of nios\_cpu module. This module contains softcore ALTERA NIOS II CPU and user accessible configuration registers for other modules. More detailed description can be found in **Table 3**. Module generic parameters are explained in **Table 4** and ports are described in **Table 5**.



Figure 3 nios\_cpu block diagram

Table 3 Description of nios\_cpu instances

| **Instance** | **Description** |
| --- | --- |
| lms\_ctr | NIOS II softcore processor instance. Processor constantly monitors input FIFO buffer connected to *exfifo\_if* ports and reads one packet containing 64 bytes. See **LMS64C control protocol** document for protocol description and command list. NIOS CPU executes received command and writes 64 bytes response packet to FIFO buffer connected to *exfifo\_of* ports. |
| cfg\_top | Wrapper module for SPI configuration registers. |
| fpgacfg | General configuration 32x16b addressable registers. Address range 0x0000 - 0x001F. See **Table 6** for register description. |
| pllcfg | PLL configuration registers. Address range 0x0020 - 0x003F. See **Table 7** for register description. |
| tstcfg | Test module configuration registers. Address range 0x0060 - 0x007F. See **Table 8** for register description. |
| txtspcfg | Configuration registers for external DAC transmit modules. Address range 0x0080 – 0x009F. See **Table 9** for register description. |
| rxtspcfg | Configuration registers for external ADC receive modules. Address range 0x00A0 – 0x00BF. See **Table 10** for register description. |
| periphcfg | Peripheral configuration registers. Address range 0x00C0 - 0x00DF. See **Table 11** for register description. |
| tamercfg | VCTCXO clock tamer configuration registers. Address range 0x00E0 – 0x00FF. See **Table 12** for register description. |
| gnsscfg | GSP module configuration registers. Address range 0x0100 – 0x011F. See **Table 13** for register description. |
| memcfg | Memory configuration registers for selecting memory modules. Only one address – 0xFFFF. Each bit enables one of multiple modules on same address. For example 0x0001 in 0xFFFF address selects first fpgacfg module for read/write, 0x0002 selects second fpgacfg module, 0x0004 – third fpgacfg module. |

Table 4 nios\_cpu module parameters

| **Parameter** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- |
| Start address of SPI registers | | | |
| FPGACFG\_START\_ADDR | integer | 0 | Start address of SPI register modules. Has to be multiple of 32 |
| PLLCFG\_START\_ADDR | integer | 32 |
| TSTCFG\_START\_ADDR | integer | 64 |
| PERIPHCFG\_START\_ADDR | integer | 192 |
| TAMERCFG\_START\_ADDR | integer | 224 |
| GNSSCFG\_START\_ADDR | integer | 256 |
| MEMCFG\_START\_ADDR | integer | 65504 |

Table 5 nios\_cpu module ports

| **Port** | **Type** | **Width** | **Description** |
| --- | --- | --- | --- |
| clk | in | 1 | Free running clock – 30.72 MHz |
| reset\_n | in | 1 | Asynchronous, active low reset |
| Control data FIFO | | | |
| exfifo\_if\_d | in | 32 | External control input FIFO data |
| exfifo\_if\_rd | out | 1 | External control input FIFO read request |
| exfifo\_if\_rdempty | in | 1 | External control input FIFO read empty |
| exfifo\_of\_d | out | 32 | External control output FIFO data |
| exfifo\_of\_wr | out | 1 | External control output FIFO write request |
| exfifo\_of\_wrfull | in | 1 | External control output FIFO write full |
| exfifo\_of\_rst | out | 1 | External control output FIFO reset request, active high |
| SPI 0 | | | |
| spi\_0\_MISO | in | 1 | SPI 0 master input |
| spi\_0\_MOSI | out | 1 | SPI 0 master output |
| spi\_0\_SCLK | out | 1 | SPI 0 clock |
| spi\_0\_SS\_n | out | 5 | SPI 0 slave select. spi\_0\_SS\_n[0] - connected to LMS7002, spi\_0\_SS\_n[1] - to internal SPI modules |
| SPI 1 | | | |
| spi\_1\_MISO | in | 1 | SPI 1 master input |
| spi\_1\_MOSI | out | 1 | SPI 1 master output |
| spi\_1\_SCLK | out | 1 | SPI 1 clock |
| spi\_1\_SS\_n | out | 2 | SPI 1 slave select. spi\_1\_SS\_n[0] - connected to onboard TCXO DAC, spi\_1\_SS\_n[1] - to phase detector ADF4002 |
| SPI 2 | | | |
| spi\_2\_MISO | in | 1 | SPI 2 master input |
| spi\_2\_MOSI | out | 1 | SPI 2 master output |
| spi\_2\_SCLK | out | 1 | SPI 2 clock |
| spi\_2\_SS\_n | out | 1 | spi\_2\_SS\_n - connected to external FPGA configuration flash |
| I2C | | | |
| i2c\_scl | inout | 1 | I2C bus clock, connected to temperature sensor and EEPROM memory. |
| i2c\_sda | inout | 1 | I2C bus data, connected to temperature sensor and EEPROM memory. |
| Genral purpose I/O | | | |
| gpi | in | 8 | Not used |
| gpo | out | 8 | gpo[0] - indicates NIOS activity. 0 - Idle, 1 - Busy.  gpo[7-1] - not used |
| LMS7002 control | | | |
| lms\_ctr\_gpio | out | 4 | lms\_ctr\_gpio[0] - LMS7002 reset. lms\_ctr\_gpio{3-1] - not used |
| VCTCXO tamer control | | | |
| vctcxo\_tune\_en | in | 1 | VCTCXO tamer enable signal |
| vctcxo\_irq | in | 1 | VCTCXO tamer interrupt signal |
| PLL reconfiguration | | | |
| pll\_rst | out | 32 | PLL reset signals. |
| pll\_rcfg\_from\_pll\_0 | in | 64 | LMS #1 TX PLL reconfiguration ports |
| pll\_rcfg\_to\_pll\_0 | out | 64 |
| pll\_rcfg\_from\_pll\_1 | in | 64 | LMS #2 TX PLL reconfiguration ports |
| pll\_rcfg\_to\_pll\_1 | out | 64 |
| pll\_rcfg\_from\_pll\_2 | in | 64 | FPGA PLL reconfiguration ports |
| pll\_rcfg\_to\_pll\_2 | out | 64 |
| pll\_rcfg\_from\_pll\_3 | in | 64 | Not used |
| pll\_rcfg\_to\_pll\_3 | out | 64 |
| pll\_rcfg\_from\_pll\_4 | in | 64 |
| pll\_rcfg\_to\_pll\_4 | out | 64 |
| pll\_rcfg\_from\_pll\_5 | in | 64 |
| pll\_rcfg\_to\_pll\_5 | out | 64 |
| Avalon Slave port 0 | | | |
| avmm\_s0\_address | in | 9 | Avalon slave port connected to phase shift modules of LMS#1. |
| avmm\_s0\_read | in | 1 |
| avmm\_s0\_readdata | out | 32 |
| avmm\_s0\_write | in | 1 |
| avmm\_s0\_writedata | in | 32 |
| avmm\_s0\_waitrequest | out | 1 |
| Avalon Slave port 1 | | | |
| avmm\_s1\_address | in | 9 | Avalon slave port connected to phase shift modules of LMS#2. |
| avmm\_s1\_read | in | 1 |
| avmm\_s1\_readdata | out | 32 |
| avmm\_s1\_write | in | 1 |
| avmm\_s1\_writedata | in | 32 |
| avmm\_s1\_waitrequest | out | 1 |
| Avalon master port 0 | | | |
| avmm\_m0\_address | out | 8 | Avalon master port connected no VCTCXO tamer module |
| avmm\_m0\_read | out | 1 |
| avmm\_m0\_waitrequest | in | 1 |
| avmm\_m0\_readdata | out | 8 |
| avmm\_m0\_readdatavalid | in | 1 |
| avmm\_m0\_write | out | 1 |
| avmm\_m0\_writedata | out | 8 |
| avmm\_m0\_clk\_clk | out | 1 |
| avmm\_m0\_reset\_reset | out | 1 |
| Configuration registers | | | |
| from\_fpgacfg\_0 | out | 512 | Input/output ports from/to SPI configuration registers |
| to\_fpgacfg\_0 | in | 512 |
| from\_fpgacfg\_1 | out | 512 |
| to\_fpgacfg\_1 | in | 512 |
| from\_fpgacfg\_2 | out | 512 |
| to\_fpgacfg\_2 | in | 512 |
| from\_pllcfg | out | 512 |
| to\_pllcfg | in | 512 |
| from\_tstcfg | out | 512 |
| to\_tstcfg | in | 512 |
| to\_tstcfg\_from\_rxtx | in | 512 |
| from\_txtspcfg\_0 | out | 512 |
| to\_txtspcfg\_0 | in | 512 |
| from\_txtspcfg\_1 | out | 512 |
| to\_txtspcfg\_1 | in | 512 |
| from\_rxtspcfg | out | 512 |
| to\_rxtspcfg | in | 512 |
| from\_periphcfg | out | 512 |
| to\_periphcfg | in | 512 |
| from\_tamercfg | out | 512 |
| to\_tamercfg | in | 512 |
| from\_gnsscfg | out | 512 |
| to\_gnsscfg | in | 512 |
| from\_memcfg | out | 512 |
| to\_memcfg | in | 512 |

### Registers of fpgacfg module

Table 6 Register description of fpgacfg module

| **Address** | **Def. value** | **Bits** | **Name** | **Description** |
| --- | --- | --- | --- | --- |
| 0x0000 |  |  | **Board identification number** | |
| 15-0 | **Board ID** | LimeSDR-PCIe (**Default 000F**) |
| 0x0001 |  | **Gateware version control** | | |
| 15-0 | **GW\_VER** | Gatewate version number |
| 0x0002 |  | **Gateware revision control** | | |
| 15-0 | **GW\_REV** | Gateware revision number |
| 0x0003 |  | **Board version control** | | |
| 15-7 | Reserved |  |
| 6-4 | **BOM\_VER** | Bill of material version |
| 3-0 | **HW\_VER** | Hardware version. |
| 0x0004 | 0000 | 15-0 | Reserved |  |
| 0x0005 | 0000 | **Clock source selection for TX and RX interfaces** | | |
| 15-2 | Reserved |  |
| 1 | **DRCT\_CLK\_EN** | RX clk: |
| 0 - PLL source **(Default)** |
| 1 - Direct clock source |
| 0 | TX clk: |
| 0 - PLL source **(Default)** |
| 1 - Direct clock source |
| 0x0006 | 0000 | 15-0 | Reserved |  |
| 0x0007 | 0303 | **RX TX MIMO Channel control** | | |
| 15-10 | Reserved |  |
| 9 | **CH\_EN** | TX ch. 1: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
| 8 | TX ch. 0: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
| 7-2 | Reserved |  |
| 1 | **CH\_EN** | RX ch. 1: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
| 0 | RX ch. 0: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
| 0x0008 | 0102 | **DIQ interface control** | | |
| 15-11 | Reserved |  |
| 10 | **DLB\_EN** | Not used |
| 9 | **SYNCH\_DIS** | Packets synchronization using timestamps: |
| 0 - Enabled |
| 1 - Disabled **(Default)** |
| 8 | **MIMO\_INT\_EN** | MIMO mode: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
| 7 | **TRIQ\_PULSE** | TRXIQ\_pulse mode: |
| 0 - OFF **(Default)** |
| 1 - ON |
| 6 | **DDR\_EN** | DIQ interface mode: |
| 0 - SDR |
| 1 - DDR **(Default)** |
| 5 | **MODE** | Limelight port mode: |
| 0 - TRXIQ **(Default)** |
| 1 - JESD207 (Currently not implemented) |
| 4-2 | Reserved |  |
| 1-0 | **SMPL\_WIDTH** | Interface sample width selection: |
| "10" - 12bit **(Default)** |
| "01" - Do not use |
| "00" - 16bit |
| 0x0009 | 0003 | **Packet control** | | |
| 15-2 | Reserved |  |
| 1 | **TXPCT\_LOSS\_CLR** | TX packets dropping flag clear: |
| 0 - Normal operation **(Default)** |
| 1 - Rising edge clears flag |
| 0 | **SMPL\_NR\_CLR** | Reset timestamp: |
| 0 - Normal operation **(Default)** |
| 1 - Timestamp is cleared |
| 0x000A | 0000 | **RX and TX module control** | | |
| 15-10 | Reserved |  |
| 9 | **TX\_PTRN\_EN** | Test pattern on TX: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 8 | **RX\_PTRN\_EN** | Test pattern on RX: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 7-2 | Reserved |  |
| 1 | **TX\_EN** | TX chain: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 0 | **RX\_EN** | RX chain: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 0x000B | 0000 | 15-0 | Reserved |  |
| 0x000C | 0003 | **WFM player control 1** | | |
| 15-2 | Reserved |  |
| 1 | **WFM\_CH\_EN** | WFM ch.1: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
| 0 | WFM ch.0: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
| 0x000D | 0001 | **WFM player control 2** | | |
| 15-3 | Reserved |  |
| 2 | **WFM\_LOAD** | WFM player file load: |
| 0 to 1 transition starts WFM file loading |
| 0 - WFM file loading disabled **(Default)** |
| 1 | **WFM\_PLAY** | WFM player loaded file play enable: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
| 0 | Reserved |  |
| 0x000E | 0002 | **WFM player control 3** | | |
| 15-2 | Reserved |  |
| 1-0 | **WFM\_SMPL\_WIDTH** | WFM player sample width control: |
| "10" - 12bit, **(Default)** |
| "01" - Do not use |
| "00" - 16bit |
| 0x000F | 0000 | 15-0 | Reserved |  |
| 0x0010 | 0000 | 15-0 | Reserved |  |
| 0x0011 | 0000 | 15-0 | Reserved |  |
| 0x0012 | FFFF | **Controlled SPI enable** | | |
| 15-8 | Reserved | Not used |
| 7 | **SPI\_SS7** |
| 6 | **SPI\_SS6** |
| 5 | **SPI\_SS5** |
| 4 | **SPI\_SS4** |
| 3 | **SPI\_SS3** |
| 2 | **SPI\_SS2** |
| 1 | **SPI\_SS1** |
| 0 | **SPI\_SS0** |
| 0x0013 | 6F6F | **LMS7002 MISC pin control** | | |
| 15 | Reserved |  |
| 14 | **LMS2\_RXEN** | Not used |
| 13 | **LMS2\_TXEN** |
| 12 | **LMS2\_TXNRX2** |
| 11 | **LMS2\_TXNRX1** |
| 10 | **LMS2\_CORE\_LDO\_EN** |
| 9 | **LMS2\_RESET** |
| 8 | **LMS2\_SS** |
| 7 | Reserved |  |
| 6 | **LMS1\_RXEN** | RX hard enable: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
| 5 | **LMS1\_TXEN** | TX hard enable: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
| 4 | **LMS1\_TXNRX2** | Port 2 mode selection: |
| 0 - TXIQ **(Default)** |
| 1 - RXIQ |
| 3 | **LMS1\_TXNRX1** | Port 1 mode selection: |
| 0 - TXIQ |
| 1 - RXIQ (**Default)** |
| 2 | **LMS1\_CORE\_LDO\_EN** | Internal LDO control: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 1 | **LMS1\_RESET** | Hardware reset: |
| 0 - Reset activated |
| 1 - Reset inactive **(Default)** |
| 0 | **LMS1\_SS** | Not used |
| 0x0014 | 0000 | 15-0 | Reserved for lms3\_4 |  |
| 0x0015 | 0000 | 15-0 | Reserved for lms5-6 |  |
| 0x0016 | 0000 | 15-0 | Reserved for lms7-8 |  |
| 0x0017 | 0000 | **GPIO for external periphery** | | |
| 15-14 | Reserved |  |
| 13 | **GPIO13** | Not used |
| 12 | **GPIO12** |
| 11 | **GPIO11** |
| 10 | **GPIO10** |
| 9 | **GPIO9** |
| 8 | **GPIO8** |
| 7 | **GPIO7** |
| 6 | **GPIO6** | Ch. B shunt: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
| 5 | **GPIO5** | Ch. B attenuator |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 4 | **GPIO4** | RF loopback ch. B: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 3 | **GPIO3** | Reserved |
| 2 | **GPIO2** | Ch. A shunt: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
| 1 | **GPIO1** | Ch. A attenuator: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 0 | **GPIO0** | RF loopback ch. A: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 0x0018 | 0001 | 15-1 | Reserved |  |
| 0 | **DEV\_CTRL0** | Not used |
| 0x0019 |  | 15-0 | Reserved |  |
| 0x001A | 0000 | **Onboard led control** | | |
| 15 | Reserved |  |
| 14 | Reserved |
| 13 | Reserved |
| 12 | Reserved |
| 11 | Reserved |
| 10 | Reserved |
| 9 | Reserveds |
| 8 | Reserved |
| 7 | Reserved |
| 6 | **FPGA\_LED2\_G** | Green LED2 control, do not turn on while red LED2 is on: |
| 0 - OFF **(Default)** |
| 1 - ON |
| 5 | **FPGA\_LED2\_R** | Red LED2 control, do not turn on while green LED2 is on: |
| 0 - OFF **(Default)** |
| 1 - ON |
| 4 | **FPGA\_LED2\_OVRD** | LED2 control override: |
| 0 - OFF **(Default)** |
| 1 - ON |
| 3 | Reserved |  |
| 2 | **FPGA\_LED1\_G** | Green LED1 control, do not turn on while red LED1 is on: |
| 0 - OFF **(Default)** |
| 1 - ON |
| 1 | **FPGA\_LED1\_R** | Red LED1 control, do not turn on while green LED1 is on: |
| 0 - OFF **(Default)** |
| 1 - ON |
| 0 | **FPGA\_LED1\_OVRD** | LED1 control override: |
| 0 - OFF **(Default)** |
| 1 - ON |
| 0x001B | 0000 | 15-8 | Reserved |  |
| 7 | Reserved |
| 6 | Reserved |
| 5 | Reserved |
| 4 | Reserved |
| 3 | Reserved |
| 2 | Reserved |
| 1 | Reserved |
| 0 | Reserved |
| 0x001C | 0000 | 15-3 | Reserved | Onboard led control |
| 2 | **FX3\_LED\_G** | Green FX3 control, do not turn on while red FX3 is on: |
| 0 - OFF **(Default)** |
| 1 - ON |
| 1 | **FX3\_LED\_R** | Red FX3 control, do not turn on while green FX3 is on: |
| 0 - OFF **(Default)** |
| 1 - ON |
| 0 | **FX3\_LED\_OVRD** | FX3 control override: |
| 0 - OFF **(Default)** |
| 1 - ON |
| 0x001D | 0000 | 15-0 | Reserved |  |
| 0x001E | 0000 | 15-0 | Reserved |  |
| 0x001F | 0000 | 15-0 | Reserved |  |

### Registers of pllcfg module

Table 7 Register description of pllcfg module

| **Address** | **Def. value** | **Bits** | **Name** | **Description** |
| --- | --- | --- | --- | --- |
| 0x0020 | 0000 | 15-0 | Reserved |  |
| 0x0021 | 0001 |  | **PLL configuration status** | |
| 15-4 | Reserved |  |
| 3 | **AUTO\_PHCFG\_ERR** | Auto phase configuration error status: |
| 0 – no error |
| 1 – Error |
| 2 | **AUTO\_PHCFG\_DONE** | Auto phase configuration status: |
| 0 – Not done |
| 1 – Done |
| 1 | **BUSY** | PLL reconfiguration busy status: |
| 0 – Idle |
| 1 – Busy |
| 0 | **DONE** | PLL configuration status: |
| 0 – Not done |
| 1 – Done |
| 0x0022 | 0000 |  | **PLL lock status** | |
| 15-2 | Reserved |  |
| 1 | **PLL\_LOCK** | RX PLL: |
| 0 – No lock |
| 1 – Locked |
| 0 | TX PLL: |
| 0 – No lock |
| 1 – Locked |
| 0x0023 | 0000 |  | **PLL control** | |
| 15 | Reserved |  |
| 14 | **PHCFG\_MODE** | PLL phase configuration mode: |
| 0 - Manual |
| 1 - AUTO |
| 13 | **PHCFG\_UpDn** | Phase shift direction: |
| 0 - Down |
| 1 - Up |
| 12-8 | **CNT\_IND** | Counter index for phase shift: |
| 0000 - All output counters |
| 0001 - M counter |
| 0010 - C0 counter |
| 0011 - C1 counter |
| 7-3 | **PLL\_IND** | PLL index for reconfiguration: |
| 0000 - TX PLL |
| 0001 - RX PLL |
| Do not use other index values |
| 2 | **PLLRST\_START** | Reset bit for PLL: |
| 0 - Reset inactive |
| 0 to 1 transition triggers reset for PLL with selected index |
| 1 | **PHCFG\_START** | Phase shift start: |
| 0 - Phase shift process inactive |
| 0 to 1 - transition triggers phase shift process for PLL with selected indexes |
| 0 | **PLLCFG\_START** | PLL reconfiguration start: |
| 0 - Phase shift process inactive |
| 0 to 1 - transition triggers phase shift process for PLL with selected indexes |
| 0x0024 | 0000 |  | **PLL reconfiguration settings** | |
| 15-0 | **CNT\_PHASE** | Counter phase value |
| 0x0025 | 01F0 | 15 | Reserved |  |
| 14-11 | **PLLCFG\_BS** | Bandwidth setting (Not used) |
| 10-8 | **CHP\_CURR** | PLL charge Pump Current (1) |
| 7 | **PLLCFG\_VCODIV** | PLL VCO division value |
| 0 = 2 |
| 1 = 1 |
| 6-2 | **PLLCFG\_LF\_RES** | PLL Loop filter resistance (1) |
| 1-0 | **PLLCFG\_LF\_CAP** | PLL Loop filter capacitance (1) |
| 0x0026 | 0001 | 15-4 | Reserved | Counter bypass and odd division control bits (1) |
| 3 | **M\_ODDDIV** |
| 2 | **M\_BYP** |
| 1 | **N\_ODDDIV** |
| 0 | **N\_BYP** |
| 0x0027 | 555A | 15 | **C7\_ODDDIV** |
| 14 | **C7\_BYP** |
| 13 | **C6\_ODDDIV** |
| 12 | **C6\_BYP** |
| 11 | **C5\_ODDDIV** |
| 10 | **C5\_BYP** |
| 9 | **C4\_ODDDIV** |
| 8 | **C4\_BYP** |
| 7 | **C3\_ODDDIV** |
| 6 | **C3\_BYP** |
| 5 | **C2\_ODDDIV** |
| 4 | **C2\_BYP** |
| 3 | **C1\_ODDDIV** |
| 2 | **C1\_BYP** |
| 1 | **C0\_ODDDIV** |
| 0 | **C0\_BYP** |
| 0x0028 | 5555 | 15 | **C15\_ODDDIV** |
| 14 | **C15\_BYP** |
| 13 | **C14\_ODDDIV** |
| 12 | **C14\_BYP** |
| 11 | **C13\_ODDDIV** |
| 10 | **C13\_BYP** |
| 9 | **C12\_ODDDIV** |
| 8 | **C12\_BYP** |
| 7 | **C11\_ODDDIV** |
| 6 | **C11\_BYP** |
| 5 | **C10\_ODDDIV** |
| 4 | **C10\_BYP** |
| 3 | **C9\_ODDDIV** |
| 2 | **C9\_BYP** |
| 1 | **C8\_ODDDIV** |
| 0 | **C8\_BYP** |
| 0x0029 |  | 15-0 | Reserved |  |
| 0x002A | 0000 | 15-8 | **N\_HCNT[15:8]** | N counter values (1) |
| 7-0 | **N\_LCNT[7:0]** |
| 0x002B | 0000 | 15-8 | **M\_HCNT[15:8]** | M counter values (1) |
| 7-0 | **M\_LCNT[7:0]** |
| 0x002C | 0000 | 15-0 | **M\_FRAC[15:0]** | M fractional counter values (Only for fractional PLL) (1) |
| 0x002D | 0000 | 15-0 | **M\_FRAC[31:16]** |
| 0x002E | 0000 | 15-8 | **C0\_HCNT[15:8]** | C0 counter values (1) |
| 7-0 | **C0\_LCNT[7:0]** |
| 0x002F | 0000 | 15-8 | **C1\_HCNT[15:8]** | C1 counter values (1) |
| 7-0 | **C1\_LCNT[7:0]** |
| 0x0030 | 0000 | 15-8 | **C2\_HCNT[15:8]** | C2counter values (1) |
| 7-0 | **C2\_LCNT[7:0]** |
| 0x0031 | 0000 | 15-8 | **C3\_HCNT[15:8]** | C3 counter values (1) |
| 7-0 | **C3\_LCNT[7:0]** |
| 0x0032 | 0000 | 15-8 | **C4\_HCNT[15:8]** | C4 counter values (1) |
| 7-0 | **C4\_LCNT[7:0]** |
| 0x0033 | 0000 | 15-8 | **C5\_HCNT[15:8]** | C5 counter values (1) |
| 7-0 | **C5\_LCNT[7:0]** |
| 0x0034 | 0000 | 15-8 | **C6\_HCNT[15:8]** | C6 counter values (1) |
| 7-0 | **C6\_LCNT[7:0]** |
| 0x0035 | 0000 | 15-8 | **C7\_HCNT[15:8]** | C7 counter values (1) |
| 7-0 | **C7\_LCNT[7:0]** |
| 0x0036 | 0000 | 15-8 | **C8\_HCNT[15:8]** | C8 counter values (1) |
| 7-0 | **C8\_LCNT[7:0]** |
| 0x0037 | 0000 | 15-8 | **C9\_HCNT[15:8]** | C9 counter values (1) |
| 7-0 | **C9\_LCNT[7:0]** |
| 0x0038 |  | 15-0 | Reserved | Reserved for C10-C15 counter values |
| 0x0039 |  | 15-0 | Reserved |
| 0x003A |  | 15-0 | Reserved |
| 0x003B |  | 15-0 | Reserved |
| 0x003C |  | 15-0 | Reserved |
| 0x003D |  | 15-0 | Reserved |
| 0x003E | 0FFF |  | **Auto phase shift options** | |
|  | **AUTO\_PHCFG\_SMPLS** | Samples to compare in auto phase shift mode |
| 0x003F | 0002 |  | **AUTO\_PHCFG\_STEP** | Step size for auto phase |

Note 1: For detailed description see “Cyclone IV Device Handbook”, Chapter 5. Clock Networks and PLLs in Cyclone IV Devices.

### Registers of tstcfg module

Table 8 Register description of tstcfg module

| **Address** | **Def. value** | **Bits** | **Type** | **Name** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 0x0060 | 00F0 |  | **SPI signature** | | |
| 15-8 |  | Reserved |  |
| 7-4 | R | **SPI\_SIGN\_REZULT** | Inverted bits from SPI\_SIGN register |
| 3-0 | R/W | **SPI\_SIGN** | SPI module test register. |
|  |  |  | **Test enable** | | |
| 0x0061 | 0000 | 15-6 |  | Reserved |  |
| 5 | R/W | **DDR2\_2\_TST\_EN** | DDR2\_2 memory test: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 4 | R/W | **DDR2\_1\_TST\_EN** | DDR2\_2 memory test: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 3 | R/W | **ADF\_TST\_EN** | Phase detector test: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 2 | R/W | **VCTCXO\_TST\_EN** | VCTCXO test: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 1 | R/W | **Si5351C\_TST\_EN** | Si5351C clock test: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 0 | R/W | **FX3\_PCLK\_TST\_EN** | FX3 PCLK clock test: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 0x0062 |  |  |  | Reserved |  |
| 0x0063 | 0000 |  | **Error insertion** | | |
| 15-6 |  | Reserved |  |
| 5 | R/W | **DDR2\_2\_TST\_FRC\_ERR** | DDR2\_2 insert error to memory test: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 4 | R/W | **DDR2\_1\_TST\_FRC\_ERR** | DDR2\_1 insert error to memory test: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 3 | R/W | **ADF\_TST\_FRC\_ERR** | Insert error to phase detector test: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 2 | R/W | **VCTCXO\_TST\_FRC\_ERR** | Insert error to VCTCXO test: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 1 | R/W | **Si5351C\_TST\_FRC\_ERR** | Insert error to Si5351C clock test: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 0 | R/W | **FX3\_PCLK\_TST\_FRC\_ERR** | Insert error to FX3 PCLK clock test: |
| 0 - Disabled **(Default)** |
| 1 - Enabled |
| 0x0064 |  |  |  | Reserved |  |
| 0x0065 | 0000 |  | **Test status** | | |
| 15-6 |  | Reserved |  |
| 5 | R | **DDR2\_2\_TST\_CMPLT** | DDR2\_2 test status: |
| 0 - Not completed |
| 1 - Completed |
| 4 | R | **DDR2\_1\_TST\_CMPLT** | DDR2\_1test status: |
| 0 - Not completed |
| 1 - Completed |
| 3 | R | **ADF\_TST\_CMPLT** | Phase detector test status: |
| 0 - Not completed |
| 1 - Completed |
| 2 | R | **VCTCXO\_TST\_CMPLT** | VCTCXO test status: |
| 0 - Not completed |
| 1 - Completed |
| 1 | R | **Si5351C\_TST\_CMPLT** | Si5351C clock test status: |
| 0 - Not completed |
| 1 - Completed |
| 0 | R | **FX3\_PCLK\_TST\_CMPLT** | FX3 PCLK clock test status: |
| 0 - Not completed |
| 1 - Completed |
| 0x0066 |  |  |  | Reserved |  |
|  |  |  | **Test results** | | |
| 0x0067 | 0000 | 15-6 |  | Reserved |  |
| 5 | R | **DDR2\_2\_TST\_REZ** | DDR2\_2 test result: |
| 0 - Fail |
| 1 - Pass |
| 4 | R | **DDR2\_1\_TST\_REZ** | DDR2\_1 test result: |
| 0 - Fail |
| 1 - Pass |
| 3 | R | **ADF\_TST\_REZ** | Not used |
| 2 | R | **VCTCXO\_TST\_REZ** | Not used |
| 1 | R | **Si5351C\_TST\_REZ** | Not used |
| 0 | R | **FX3\_PCLK\_TST\_REZ** | Not used |
|  |  |  | Clock test counter values | | |
| 0x0068 |  |  |  | Reserved |  |
| 0x0069 |  |  | R | **FX3\_CLK\_CNT** | FX3 PCLK clock counter value |
| 0x006A |  |  | R | **Si5351C\_CLK0\_CNT** | Si5351C CLK0 counter value |
| 0x006B |  |  | R | **Si5351C\_CLK1\_CNT** | Si5351C CLK1 counter value |
| 0x006C |  |  | R | **Si5351C\_CLK2\_CNT** | Si5351C CLK2 counter value |
| 0x006D |  |  | R | **Si5351C\_CLK3\_CNT** | Si5351C CLK3 counter value |
| 0x006E |  |  |  | Reserved |  |
| 0x006F |  |  | R | **Si5351C\_CLK5\_CNT** | Si5351C CLK5 counter value |
| 0x0070 |  |  | R | **Si5351C\_CLK6\_CNT** | Si5351C CLK6 counter value |
| 0x0071 |  |  | R | **Si5351C\_CLK7\_CNT** | Si5351C CLK7 counter value |
| 0x0072 |  |  | R | **LMK\_CLK\_CNT\_L** | LMK clock counter value |
| 0x0073 |  |  | R | **LMK\_CLK\_CNT\_H** |
| 0x0074 |  |  | R | **ADF\_CNT** | ADF transition count value |
| 0x0075 |  |  |  | Reserved |  |
|  |  |  | **DDR2\_1 detailed test results 1** | | |
| 0x0076 |  | 15-3 |  | Reserved |  |
| 2 | R | **DDR2\_1\_TST\_FAIL** | DDR2\_1 test result: |
| 0 - Test not completed |
| 1 - Fail |
| 1 | R | **DDR2\_1\_TST\_PASS** | DDR2\_1 test result: |
| 0 - Test not completed |
| 1 - Pass |
| 0 | R | **DDR2\_1\_TST\_CMPLT** | DDR2\_1 test result: |
| 0 - Test not completed |
| 1 - Test complete |
| 0x0077 |  |  | **DDR2\_1 detailed test results 2** | | |
| 15-0 | R | **DDR2\_1\_PNF\_PER\_BIT\_L** | DDR2\_1 data [15:0] bus pas not fail per bit: |
| 0 - Fail |
| 1 - Pass |
| 0x0078 |  |  | **DDR2\_1 detailed test results 3** | | |
| 15-0 | R | **DDR2\_1\_PNF\_PER\_BIT\_H** | DDR2\_1 data [31:16] bus pas not fail per bit: |
| 0 - Fail |
| 1 - Pass |
| 0x0079 |  | 15-0 |  | Reserved |  |
| 0x007A |  |  | **DDR2\_2 detailed test results 1** | | |
| 15-3 |  | Reserved |  |
| 2 | R | **DDR2\_2\_TST\_FAIL** | DDR2\_2 test result: |
| 0 - Test not completed |
| 1 - Fail |
| 1 | R | **DDR2\_2\_TST\_PASS** | DDR2\_2 test result: |
| 0 - Test not completed |
| 1 - Pass |
| 0 | R | **DDR2\_2\_TST\_CMPLT** | DDR2\_2 test result: |
| 0 - Test not completed |
| 1 - Test complete |
| 0x007B |  |  | **DDR2\_2 detailed test results 2** | | |
| 15-0 | R | **DDR2\_2\_PNF\_PER\_BIT\_L** | DDR2\_2 data [15:0] bus pas not fail per bit: |
| 0 - Fail |
| 1 - Pass |
| 0x007C |  |  | **DDR2\_2 detailed test results 3** | | |
| 15-0 | R | **DDR2\_2\_PNF\_PER\_BIT\_H** | DDR2\_2 data [31:16] bus pas not fail per bit: |
| 0 - Fail |
| 1 - Pass |
| 0x007D | AAAA |  | **TX test pattern 1** | | |
| 15-0 | R/W | **TX\_TST\_I** | TX test pattern I sample value |
| 0x007E | 5555 |  | **TX test pattern 2** | | |
| 15-0 | R/W | **TX\_TST\_Q** | TX test pattern Q sample value |
| 0x007F |  | 15-0 |  | Reserved |  |

### Registers of txtspcfg module

Table 9 regsiters of txtspcfg module

| **Address** | **Def. value** | **Bits** | **Type** | **Name** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 0x0080 | 0081 |  | **Control** | | |
| 15-8 |  | Reserved |  |
| 2 | R/W | **INSEL** | Input select: |
| 0 - TX path **(Default)** |
| 1 - NCO |
| 1 | R/W | Reserved |  |
| 0 | R/W | **EN** | Module enable: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
|  |  |  | **Corrector values** | | |
| 0x0081 | 07FF | 15-6 |  | Reserved |  |
| 10-0 | R/W | **gcorrQ** | Q gain corrector value |
| 0x0082 | 07FF | 15-6 |  | Reserved |  |
| 10-0 | R/W | **gcorrI** | I gain corrector value |
| 0x0083 | 0000 | 15-12 |  | Reserved |  |
| 11-0 | R/W | **IQcorr** | IQ corrector value |
| 0x0084 | 0000 | 15-8 | R/W | **dccorrI** | DC corrector values |
| 7-0 | R/W | **dccorrQ** |
| 0x0085 |  |  |  | Reserved |  |
| 0x0086 |  |  |  | Reserved |  |
| 0x0087 |  |  |  | Reserved |  |
|  |  |  | **Correctro bypass** | | |
| 0x0088 | 0000 | 15-4 |  | Reserved |  |
| 3 | R/W | **DC\_BYP** | DC corrector bypass: |
| 0 - Not bypassed **(Default)** |
| 1 - Bypassed |
| 2 |  | Reserved |  |
| 1 | R/W | **GC\_BYP** | Gain corrector bypass: |
| 0 - Not bypassed **(Default)** |
| 1 - Bypassed |
| 0 | R/W | **PH\_BYP** | IQ corrector bypass: |
| 0 - Not bypassed **(Default)** |
| 1 - Bypassed |
| 0x0089 |  |  |  | Reserved |  |
| 0x008A |  |  |  | Reserved |  |
| 0x008B |  |  |  | Reserved |  |
| 0x008C |  |  |  | Reserved |  |
| 0x008D |  |  |  | Reserved |  |
| 0x008E |  |  |  | Reserved |  |
| 0x008F |  |  |  | Reserved |  |
| 0x0090 |  |  |  | Reserved |  |
| 0x0091 |  |  |  | Reserved |  |
| 0x0092 |  |  |  | Reserved |  |
| 0x0093 |  |  |  | Reserved |  |
| 0x0094 |  |  |  | Reserved |  |
| 0x0095 |  |  |  | Reserved |  |
| 0x0096 |  |  |  | Reserved |  |
| 0x0097 |  |  |  | Reserved |  |
| 0x0098 |  |  |  | Reserved |  |
| 0x0099 |  |  |  | Reserved |  |
| 0x009A |  |  |  | Reserved |  |
| 0x009B |  |  |  | Reserved |  |
| 0x009C |  |  |  | Reserved |  |
| 0x009D |  |  |  | Reserved |  |
| 0x009E |  |  |  | Reserved |  |
| 0x009F |  |  |  | Reserved |  |

### Registers of rxtspcfg module

Table 10 regsiters of rxtspcfg module

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Address** | **Def. value** | **Bits** | **Type** | **Name** | **Description** |
| 0x00A0 | 0081 |  | **Control** | | |
| 15-1 |  | Reserved |  |
| 0 | R/W | **EN** | Module enable: |
| 0 - Disabled |
| 1 - Enabled **(Default)** |
|  |  |  | **Corrector values** | | |
| 0x00A1 | 07FF | 15-6 |  | Reserved |  |
| 10-0 | R/W | **gcorrQ** | Q gain corrector value |
| 0x00A2 | 07FF | 15-6 |  | Reserved |  |
| 10-0 | R/W | **gcorrI** | I gain corrector value |
| 0x00A3 | 0000 | 15-12 |  | Reserved |  |
| 11-0 | R/W | **IQcorr** | IQ corrector value |
| 0x00A4 | 0000 | 15-3 | R/W | Reserved |  |
| 2-0 | R/W | **DCCORR\_AVG** | DC corrector value |
| 0x00A5 |  |  |  | Reserved |  |
| 0x00A6 |  |  |  | Reserved |  |
| 0x00A7 |  |  |  | Reserved |  |
| 0x00A8 |  |  |  | Reserved |  |
| 0x00A9 |  |  |  | Reserved |  |
| 0x00AA |  |  |  | Reserved |  |
| 0x00AB |  |  |  | Reserved |  |
|  |  |  | **Corrector bypass** | | |
| 0x00AC | 0000 | 15-3 |  | Reserved |  |
| 2 | R/W | **DC\_BYP** | DC corrector bypass: |
| 0 - Not bypassed **(Default)** |
| 1 - Bypassed |
| 1 | R/W | **GC\_BYP** | Gain corrector bypass: |
| 0 - Not bypassed **(Default)** |
| 1 - Bypassed |
| 0 | R/W | **PH\_BYP** | IQ corrector bypass: |
| 0 - Not bypassed **(Default)** |
| 1 - Bypassed |
| 0x00AD |  |  |  | Reserved |  |
| 0x00AE |  |  |  | Reserved |  |
| 0x00AF |  |  |  | Reserved |  |
| 0x00B0 |  |  |  | Reserved |  |
| 0x00B1 |  |  |  | Reserved |  |
| 0x00B2 |  |  |  | Reserved |  |
| 0x00B3 |  |  |  | Reserved |  |
| 0x00B4 |  |  |  | Reserved |  |
| 0x00B5 |  |  |  | Reserved |  |
| 0x00B6 |  |  |  | Reserved |  |
| 0x00B7 |  |  |  | Reserved |  |
| 0x00B8 |  |  |  | Reserved |  |
| 0x00B9 |  |  |  | Reserved |  |
| 0x00BA |  |  |  | Reserved |  |
| 0x00BB |  |  |  | Reserved |  |
| 0x00BC |  |  |  | Reserved |  |
| 0x00BD |  |  |  | Reserved |  |
| 0x00BE |  |  |  | Reserved |  |
| 0x00BF |  |  |  | Reserved |  |

### Registers of periphcfg module

Table 11 Register description of periphcfg module

| **Address** | **Def. value** | **Bits** | **Type** | **Name** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 0x00C0 | FFFF |  | **Board GPIO control 1** | | |
| 15-8 |  | Reserved |  |
| 7-0 | **R/W** | **BOARD\_GPIO\_OVRD** | GPIO control override (each bit controls corresponding GPIO): |
| 0 - Dedicated function |
| 1 - Overridden by user **(Default)** |
| 0x00C1 |  | 15-0 |  | Reserved for GPIO |  |
| 0x00C2 | 0000 |  | **Board GPIO control 2** | | |
| 15-8 |  | Reserved |  |
| 7-0 | **R** | **BOARD\_GPIO\_RD** | GPIO read value (each from corresponding GPIO): |
| 0 - Low level |
| 1 - High level |
| 0x00C3 |  | 15-0 |  | Reserved for GPIO |  |
| 0x00C4 | 0000 |  | **Board GPIO control 3** | | |
| 15-8 |  | Reserved |  |
| 7-0 | **R/W** | **BOARD\_GPIO\_DIR** | Onboard GPIO direction (each bit controls corresponding GPIO): |
| 0 - Input **(Default)** |
| 1 - Output |
| 0x00C5 |  | 15-0 |  | Reserved for GPIO |  |
| 0x00C6 | 0000 |  | **Board GPIO control 4** | | |
| 15-8 |  | Reserved |  |
| 7-0 | **R/W** | **BOARD\_GPIO\_VAL** | GPIO output value (each bit controls corresponding GPIO): |
| 0 - Low level |
| 1 - High level |
| 0x00C7 |  | 15-0 |  | Reserved for GPIO |  |
| 0x00C8 | 0000 | 15-0 |  | **PERIPH\_INPUT\_RD\_0** | Not used |
| 0x00C9 | 0000 | 15-0 |  | **PERIPH\_INPUT\_RD\_1** | Not used |
| 0x00CA |  | 15-0 |  | Reserved |  |
| 0x00CB |  | 15-0 |  | Reserved |  |
| 0x00CC | 0000 |  | **Board peripheral control 1** | | |
| 15-1 |  | **PERIPH\_OUTPUT\_OVRD\_0** | Not used |
| 0 | **R/W** | Fan control override: |
| 0 - Dedicated function **(Default)** |
| 1 - User controlled |
| 0x00CD | 0000 |  | **Board peripheral control 1** | | |
| 15-1 |  | **PERIPH\_OUTPUT\_VAL\_0** | Not used |
| 0 | **R/W** | Fan control pin: |
| 0 - OFF **(Default)** |
| 1- ON |
| 0x00CE | 0000 | 15-0 |  | **PERIPH\_OUTPUT\_OVRD\_1** | Not used |
| 0x00CF | 0000 | 15-0 |  | **PERIPH\_OUTPUT\_VAL\_1** | Not used |
| 0x00D0 |  | 15-0 |  | Reserved |  |
| 0x00D1 |  | 15-0 |  | Reserved |  |
| 0x00D2 |  | 15-0 |  | Reserved |  |
| 0x00D3 |  | 15-0 |  | Reserved |  |
| 0x00D4 |  | 15-0 |  | Reserved |  |
| 0x00D5 |  | 15-0 |  | Reserved |  |
| 0x00D6 |  | 15-0 |  | Reserved |  |
| 0x00D7 |  | 15-0 |  | Reserved |  |
| 0x00D8 |  | 15-0 |  | Reserved |  |
| 0x00D9 |  | 15-0 |  | Reserved |  |
| 0x00DA |  | 15-0 |  | Reserved |  |
| 0x00DB |  | 15-0 |  | Reserved |  |
| 0x00DC |  | 15-0 |  | Reserved |  |
| 0x00DD |  | 15-0 |  | Reserved |  |
| 0x00DE |  | 15-0 |  | Reserved |  |
| 0x00DF |  | 15-0 |  | Reserved |  |

### Registers of vctcxocfg module

Table 12 Registers of vctcxocfg module

| **Address** | **Def. value** | **Bits** | **Type** | **Name** | **Description** |
| --- | --- | --- | --- | --- | --- |
| 0x00E0 | 0000 |  | **Control** | | |
| 15-1 |  | **Reserved** |  |
| 0 | R/W | **EN** | 1 - Enabled, 0 – Disabled |
|  |  |  | **Status** | | |
| 0x00E1 | 0000 | 15-8 |  | **Reserved** |  |
| 7-4 | R | **ACCURACY** | “0000” - tune disabled or lowest accuracy, 0001 – 1s tune , 0010 – 2s tune , 0011 – 3s tune (highest accuracy). |
| 3-0 | R | **STATE** | “0000” - Coarse Tune, “0001” - Fine tune |
| 0x00E2 | 0000 | 15-0 | R | **DAC\_TUNED\_VAL** | DAC tuned value to get frequency with best accuracy |
|  |  |  | **Tune settings** | | |
| 0x00E3 | - | 15-0 | R/W | **PPS\_1S\_ERR\_TOL\_L** | Error tolerance value in 1s period (32 bit value, L – lower 16 b, H – upper 16b). Default values are board dependant. |
| 0x00E4 | - | 15-0 | R/W | **PPS\_1S\_ERR\_TOL\_H** |
| 0x00E5 | - | 15-0 | R/W | **PPS\_10S\_ERR\_TOL\_L** | Error tolerance value in 10s period (32 bit value, L – lower 16 b, H – upper 16b). Default values are board dependant. |
| 0x00E6 | - | 15-0 | R/W | **PPS\_10S\_ERR\_TOL\_H** |
| 0x00E7 | - | 15-0 | R/W | **PPS\_100S\_ERR\_TOL\_L** | Error tolerance value in 100s period (32 bit value, L – lower 16 b, H – upper 16b). Default values are board dependant. |
| 0x00E8 | - | 15-0 | R/W | **PPS\_100S\_ERR\_TOL\_H** |
|  |  |  | **Error values** | | |
| 0x00E9 | 0000 | 15-0 | R | **PPS\_1S\_ERR\_L** | Error count in 1s period (32 bit signed value, L – lower 16 b, H – upper 16b) |
| 0x00EA | 0000 | 15-0 | R | **PPS\_1S\_ERR\_H** |
| 0x00EB | 0000 | 15-0 | R | **PPS\_10S\_ERR\_L** | Error count in 10s period (32 bit signed value, L – lower 16 b, H – upper 16b) |
| 0x00EC | 0000 | 15-0 | R | **PPS\_10S\_ERR\_H** |
| 0x00ED | 0000 | 15-0 | R | **PPS\_100S\_ERR\_L** | Error count in 100s period (32 bit signed value, L – lower 16 b, H – upper 16b) |
| 0x00EE | 0000 | 15-0 | R | **PPS\_100S\_ERR\_H** |

### Registers of gnsscfg module

Table 13 Registers of gnsscfg module

| **Address** | **Def. value** | **Bits** | **Type** | **Name** | **Description** |
| --- | --- | --- | --- | --- | --- |
|  |  |  | **Control** | | |
| 0x0100 | s0000 | 15-1 |  | **Reserved** |  |
| 0 | R/W | **EN** | 1 - Enabled, 0 – Disabled |
|  |  |  | **Status** | | |
| 0x0101 | 0000 | 15-12 |  | **Reserved** | UTC of position fix (BCD format). HH-MM-SS1.SSS0 |
| 11-0 | R | **GPRMC\_UTC\_SSS0** |
| 0x0102 | 0000 | 15-8 | R | **GPRMC\_UTC\_MM** |
| 7-0 | R | **GPRMC\_UTC\_SS1** |
| 0x0103 | 0000 | 15-8 |  | **Reserved** |
| 7-0 | R | **GPRMC\_UTC\_HH** |
| 0x0104 | 0000 | 15-1 |  | **Reserved** |  |
| 0 | R | **GPRMC\_STATUS** | Status 1 = Data valid, 0 = Navigation receiver warning |
| 0x0105 | 0000 | 15-8 | R | **GPRMC\_LAT\_LL1** | Latitude,**LL3-LL2.LL1-LL0** |
| 7-0 | R | **GPRMC\_LAT\_LL0** |
| 0x0106 | 0000 | 15-8 | R | **GPRMC\_LAT\_LL3** |
| 7-0 | R | **GPRMC\_LAT\_LL2** |
| 0x0107 | 0000 | 15-1 |  | **Reserved** |  |
| 0 | R | **GPRMC\_LAT\_N\_S** | Latitude 0 – N, 1 – S |
| 0x0108 | 0000 | 15-8 | R | **GPRMC\_LONG\_YY1** | Longitude,**Y4-YY3-YY2.YY1-YY0** |
| 7-0 | R | **GPRMC\_LONG\_YY0** |
| 0x0109 | 0000 | 15-8 | R | **GPRMC\_LONG\_YY3** |
| 7-0 | R | **GPRMC\_LONG\_YY2** |
| 0x010A | 0000 | 15-4 |  | **Reserved** |
| 3-0 | R | **GPRMC\_LONG\_Y4** |
| 0x010B | 0000 | 15-1 |  | **Reserved** |  |
| 0 | R | **GPRMC\_LONG\_E\_W** | Longitude, 0 – E, 1 – W |
| 0x010C | 0000 | 15-8 | R | **GPRMC\_SPEED\_XX1** | Speed over ground, knots,**XX2-XX1.XX0** |
| 7-0 | R | **GPRMC\_SPEED\_XX0** |
| 0x010D | 0000 | 15-8 |  | **Reserved** |
| 7-0 | R | **GPRMC\_SPEED\_XX2** |
| 0x010E | 0000 | 15-8 | R | **GPRMC\_COURSE\_XX1** | Course Over Ground, degrees True,**X2-XX1.XX0** |
| 7-0 | R | **GPRMC\_COURSE\_XX0** |
| 0x010F | 0000 | 15-4 |  | **Reserved** |
| 3-0 | R | **GPRMC\_COURSE\_X2** |
| 0x0110 | 0000 | 15-8 | R | **GPRMC\_DATE\_MM** | Date:**DD-MM-YY** |
| 7-0 | R | **GPRMC\_DATE\_YY** |
| 0x0111 | 0000 | 15-8 |  | **Reserved** |
| 7-0 | R | **GPRMC\_DATE\_DD** |
| 0x0112 | 0000 | 15-0 |  | **Reserved** |  |
| 0x0113 | 0000 | 15-0 |  | **Reserved** |  |
| 0x0114 | 0000 | 15-8 |  | **Reserved** |  |
| 7-4 | R | **GPGSA\_FIX** | 1 = Fix not available, 2 = 2D, 3 = 3D |
| 3-0 |  | **Reserved** |  |

## PCIe interface – pcie\_top

Provides data transfer between external host and FPGA trough PCIe interface.

All data exchange between pcie\_top module and other FPGA logic is done through FIFO buffers. Module xillybus constantly monitors all FIFO buffers. For example, internal logic writes IQ stream packets containing 4kB data to FIFO buffer through F2H\_S0 ports. Once xillybus module detects that F2H\_S0 FIFO buffer is not empty and external host is ready, all data is read from FIFO buffer and written to host controller trough PCIe interface.



Figure 4 pcie\_top block diagram

Table 14 Description of pcie\_top instances

| **Instance** | **Description** |
| --- | --- |
| xillybus | Provides data transfer between PCIe interface and internal FIFO buffers. |
| fifo\_inst (F2H\_S0) | Stream 0 endpoint FIFO buffer of 8kB size. |
| fifo\_inst (H2F\_S0\_0) | Stream 0 endpoint FIFO buffer of 4kB size. |
| fifo\_inst (H2F \_S0\_1) | Stream 0 endpoint FIFO buffer of 4kB size. |
| fifo\_inst (F2H\_S1) | Stream 1 endpoint FIFO buffer of 8kB size. |
| fifo\_inst (H2F \_S1\_0) | Stream 1 endpoint FIFO buffer of 4kB size. |
| fifo\_inst (H2F \_S1\_1) | Stream 1 endpoint FIFO buffer of 4kB size. |
| fifo\_inst (F2H\_S2) | Stream 2 endpoint FIFO buffer of 8kB size. |
| fifo\_inst (H2F \_S2\_0) | Stream 2 endpoint FIFO buffer of 4kB size. |
| fifo\_inst (H2F \_S2\_1) | Stream 2 endpoint FIFO buffer of 4kB size. |
| fifo\_inst (F2H\_C0) | Control endpoint FIFO buffer of 1kB size. |
| fifo\_inst (H2F\_C0) | Control endpoint FIFO buffer of 1kB size. |

Table 15 pcie\_top module parameters

| **Parameter** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- |
| General parameters | | | |
| dev\_family | string | Cyclone V GX | Device family name |
| g\_S0\_DATA\_WIDTH | integer | 32 | Stream 0 interface data width |
| g\_S1\_DATA\_WIDTH | integer | 32 | Stream 1 interface data width |
| g\_S2\_DATA\_WIDTH | integer | 32 | Stream 2 interface data width |
| g\_C0\_DATA\_WIDTH | integer | 8 | Control interface data width |
| Stream (PC->FPGA) | | | |
| g\_H2F\_S0\_0\_RDUSEDW\_WIDTH | integer | 9 | Host to FPGA stream FIFO read used words size (29-1= 256 words) |
| g\_H2F\_S0\_0\_RWIDTH | integer | 128 | Host to FPGA stream FIFO read word size |
| g\_H2F\_S0\_1\_RDUSEDW\_WIDTH | integer | 10 | Host to FPGA stream FIFO read used words size (210-1= 512 words) |
| g\_H2F\_S0\_1\_RWIDTH | integer | 64 | Host to FPGA stream FIFO read word size |
| g\_H2F\_S1\_0\_RDUSEDW\_WIDTH | integer | 9 | Host to FPGA stream FIFO read used words size (29-1= 256 words) |
| g\_H2F\_S1\_0\_RWIDTH | integer | 128 | Host to FPGA stream FIFO read word size |
| g\_H2F\_S1\_1\_RDUSEDW\_WIDTH | integer | 10 | Host to FPGA stream FIFO read used words size (210-1= 512 words) |
| g\_H2F\_S1\_1\_RWIDTH | integer | 64 | Host to FPGA stream FIFO read word size |
| g\_H2F\_S2\_0\_RDUSEDW\_WIDTH | integer | 9 | Host to FPGA stream FIFO read used words size (29-1= 256 words) |
| g\_H2F\_S2\_0\_RWIDTH | integer | 128 | Host to FPGA stream FIFO read word size |
| g\_H2F\_S2\_1\_RDUSEDW\_WIDTH | integer | 10 | Host to FPGA stream FIFO read used words size (210-1= 512 words) |
| g\_H2F\_S2\_1\_RWIDTH | integer | 64 | Host to FPGA stream FIFO read word size |
| Stream (FPGA->PC) | | | |
| g\_F2H\_S0\_WRUSEDW\_WIDTH | integer | 11 | FPGA to host stream FIFO write used words size (211-1= 1028 words) |
| g\_F2H\_S0\_WWIDTH | integer | 64 | FPGA to host stream FIFO write word size |
| g\_F2H\_S1\_WRUSEDW\_WIDTH | integer | 11 | FPGA to host stream FIFO write used words size (211-1= 1028 words) |
| g\_F2H\_S1\_WWIDTH | integer | 64 | FPGA to host stream FIFO write word size |
| g\_F2H\_S2\_WRUSEDW\_WIDTH | integer | 11 | FPGA to host stream FIFO write used words size (211-1= 1028 words) |
| g\_F2H\_S2\_WWIDTH | integer | 64 | FPGA to host stream FIFO write word size |
| Control (PC->FPGA) | | | |
| g\_H2F\_C0\_RDUSEDW\_WIDTH | integer | 11 | Host to FPGA control FIFO read used words size (211-1= 1024 words) |
| g\_H2F\_C0\_RWIDTH | integer | 8 | Host to FPGA control FIFO read word size |
| Control (FPGA->PC) | | | |
| g\_F2H\_C0\_WRUSEDW\_WIDTH | integer | 11 | FPGA to host control FIFO write used words size (211-1= 1024 words) |
| g\_F2H\_C0\_WWIDTH | integer | 8 | FPGA to host control FIFO write word size |

Table 16 pcie\_top module ports

| **Port** | **Type** | **Width** | **Description** |
| --- | --- | --- | --- |
| clk | in | 1 | Clock 100 MHz |
| reset\_n | in | 1 | Reset active low |
| PCIe interface | | | |
| pcie\_perstn | in | 1 | Link reactivation |
| pcie\_refclk | in | 1 | Reference clock |
| pcie\_rx | in | 4 | PCIe Receive ports |
| pcie\_tx | out | 4 | PCIe Transmit ports |
| pcie\_bus\_clk | out | 1 | PCIe bus user interface clock |
| H2F\_S0 buffer select | | | |
| H2F\_S0\_0\_sel | in | 1 | 0 - H2F\_S0\_0, 1 - H2F\_S0\_1 |
| H2F\_S1\_0\_sel | in | 1 | 0 - H2F\_S1\_0, 1 - H2F\_S1\_1 |
| H2F\_S2\_0\_sel | in | 1 | 0 - H2F\_S1\_0, 1 - H2F\_S1\_1 |
| Stream 0 endpoint FIFO 0 (Host->FPGA) | | | |
| H2F\_S0\_0\_rdclk | in | 1 | Read clock |
| H2F\_S0\_0\_aclrn | in | 1 | Asynchronous clear, active low |
| H2F\_S0\_0\_rd | in | 1 | Read request |
| H2F\_S0\_0\_rdata | out | g\_H2F\_S0\_0\_RWIDTH | Read data |
| H2F\_S0\_0\_rempty | out | 1 | Read empty |
| H2F\_S0\_0\_rdusedw | out | g\_H2F\_S0\_0\_RDUSEDW\_WIDTH | Red used words |
| Stream 0 endpoint FIFO 1 (Host->FPGA) | | | |
| H2F\_S0\_1\_rdclk | in | 1 | Read clock |
| H2F\_S0\_1\_aclrn | in | 1 | Asynchronous clear, active low |
| H2F\_S0\_1\_rd | in | 1 | Read request |
| H2F\_S0\_1\_rdata | out | g\_H2F\_S0\_1\_RWIDTH | Read data |
| H2F\_S0\_1\_rempty | out | 1 | Read empty |
| H2F\_S0\_1\_rdusedw | out | g\_H2F\_S0\_1\_RDUSEDW\_WIDTH | Red used words |
| Stream 1 endpoint FIFO 0 (Host->FPGA) | | | |
| H2F\_S1\_0\_rdclk | in | 1 | Read clock |
| H2F\_S1\_0\_aclrn | in | 1 | Asynchronous clear, active low |
| H2F\_S1\_0\_rd | in | 1 | Read request |
| H2F\_S1\_0\_rdata | out | g\_H2F\_S1\_0\_RWIDTH | Read data |
| H2F\_S1\_0\_rempty | out | 1 | Read empty |
| H2F\_S1\_0\_rdusedw | out | g\_H2F\_S1\_0\_RDUSEDW\_WIDTH | Red used words |
| Stream 1 endpoint FIFO 1 (Host->FPGA) | | | |
| H2F\_S1\_1\_rdclk | in | 1 | Read clock |
| H2F\_S1\_1\_aclrn | in | 1 | Asynchronous clear, active low |
| H2F\_S1\_1\_rd | in | 1 | Read request |
| H2F\_S1\_1\_rdata | out | g\_H2F\_S1\_1\_RWIDTH | Read data |
| H2F\_S1\_1\_rempty | out | 1 | Read empty |
| H2F\_S1\_1\_rdusedw | out | g\_H2F\_S1\_1\_RDUSEDW\_WIDTH | Red used words |
| Stream 2 endpoint FIFO 0 (Host->FPGA) | | | |
| H2F\_S2\_0\_rdclk | in | 1 | Read clock |
| H2F\_S2\_0\_aclrn | in | 1 | Asynchronous clear, active low |
| H2F\_S2\_0\_rd | in | 1 | Read request |
| H2F\_S2\_0\_rdata | out | g\_H2F\_S2\_0\_RWIDTH | Read data |
| H2F\_S2\_0\_rempty | out | 1 | Read empty |
| H2F\_S2\_0\_rdusedw | out | g\_H2F\_S2\_0\_RDUSEDW\_WIDTH | Red used words |
| Stream 2 endpoint FIFO 1 (Host->FPGA) | | | |
| H2F\_S2\_1\_rdclk | in | 1 | Read clock |
| H2F\_S2\_1\_aclrn | in | 1 | Asynchronous clear, active low |
| H2F\_S2\_1\_rd | in | 1 | Read request |
| H2F\_S2\_1\_rdata | out | g\_H2F\_S2\_1\_RWIDTH | Read data |
| H2F\_S2\_1\_rempty | out | 1 | Read empty |
| H2F\_S2s\_1\_rdusedw | out | g\_H2F\_S2\_1\_RDUSEDW\_WIDTH | Red used words |
| Stream 0 endpoint FIFO (FPGA->Host) | | | |
| F2H\_S0\_wclk | in | 1 | Write clock |
| F2H\_S0\_aclrn | in | 1 | Asynchronous clear, active low |
| F2H\_S0\_wr | in | 1 | Write request |
| F2H\_S0\_wdata | in | g\_F2H\_S0\_WWIDTH | Write data |
| F2H\_S0\_wfull | out | 1 | Write full |
| F2H\_S0\_wrusedw | out | g\_F2H\_S0\_WRUSEDW\_WIDTH | Write used words |
| Stream 1 endpoint FIFO (FPGA->Host) | | | |
| F2H\_S1\_wclk | in | 1 | Write clock |
| F2H\_S1\_aclrn | in | 1 | Asynchronous clear, active low |
| F2H\_S1\_wr | in | 1 | Write request |
| F2H\_S1\_wdata | in | g\_F2H\_S1\_WWIDTH | Write data |
| F2H\_S1\_wfull | out | 1 | Write full |
| F2H\_S1\_wrusedw | out | g\_F2H\_S1\_WRUSEDW\_WIDTH | Write used words |
| Stream 2 endpoint FIFO (FPGA->Host) | | | |
| F2H\_S2\_wclk | in | 1 | Write clock |
| F2H\_S2\_aclrn | in | 1 | Asynchronous clear, active low |
| F2H\_S2\_wr | in | 1 | Write request |
| F2H\_S2\_wdata | in | g\_F2H\_S2\_WWIDTH | Write data |
| F2H\_S2\_wfull | out | 1 | Write full |
| F2H\_S2\_wrusedw | out | g\_F2H\_S2\_WRUSEDW\_WIDTH | Write used words |
| Control endpoint FIFO (Host->FPGA) | | | |
| H2F\_C0\_rdclk | in | 1 | Read clock |
| H2F\_C0\_aclrn | in | 1 | Asynchronous clear, active low |
| H2F\_C0\_rd | in | 1 | Read request |
| H2F\_C0\_rdata | out | g\_H2F\_C0\_RWIDTH | Read data |
| H2F\_C0\_rempty | out | 1 | Read empty |
| Control endpoint FIFO (FPGA->Host) | | | |
| F2H\_C0\_wclk | in | 1 | Write clock |
| F2H\_C0\_aclrn | in | 1 | Asynchronous clear, active low |
| F2H\_C0\_wr | in | 1 | Write request |
| F2H\_C0\_wdata | in | g\_F2H\_C0\_WWIDTH | Write data |
| F2H\_C0\_wfull | out | 1 | Write full |
| Status | | | |
| F2H\_S0\_open | out | 1 | Indicates when stream endpoint S0 is opened from host. |
| F2H\_S1\_open | out | 1 | Indicates when stream endpoint S1 s opened from host. |
| F2H\_S2\_open | out | 1 | Indicates when stream endpoint S2 s opened from host. |

## Packet receive and transmit interface – rxtx\_top

Main function of rxtx\_top module is for receive and transmit IQ sample packets from/to LMS7002 chip and provide IQ sample synchronization. See **Figure 5** for block diagram and **Table 17** for instance description.



Figure 5 rxtx\_top block diagram

Table 17 Description of rxtx\_top instances

| **Instance** | **Description** |
| --- | --- |
| tx\_path\_top | Transmit logic. See **3.5.2 Transmit interface – tx\_path\_top**. |
| rx\_path\_top | Receive logic. See **3.5.1 Receive interface – rx\_path\_top**. |

Table 18 rxtx\_top parameters description

| **Parameter** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- |
| DEV\_FAMILY | string | Cyclone V | Device family |
| TX parameters | | | |
| TX\_IQ\_WIDTH | integer | 12 | TX IQ sample width |
| TX\_N\_BUFF | integer | 2 | TX number of buffers, 2,4 valid values |
| TX\_IN\_PCT\_SIZE | integer | 4096 | TX packet size in bytes |
| TX\_IN\_PCT\_HDR\_SIZE | integer | 16 | TX packet header size in bytes |
| TX\_IN\_PCT\_DATA\_W | integer | 128 | TX packet read data width |
| TX\_IN\_PCT\_RDUSEDW\_W | integer | 11 | TX packet read used words width |
| TX\_OUT\_PCT\_DATA\_W | integer | 64 | TX output packet data width |
| TX\_SMPL\_FIFO\_WRUSEDW\_W | integer | 9 | TX sample FIFO write used words |
| RX parameters | | | |
| RX\_IQ\_WIDTH | integer | 12 | RX IQ sample width |
| RX\_INVERT\_INPUT\_CLOCKS | string | ON | Clock invert option on LMS\_DIQ2 interface |
| RX\_SMPL\_BUFF\_RDUSEDW\_W | integer | 11 | RX sample buffer read used words width. Words=211-1 |
| RX\_PCT\_BUFF\_WRUSEDW\_W | integer | 11 | RX packet buffer read used words width. Words=211-1 |

Table 19 rxtx\_top port description

| **Port** | **Type** | **Width** | **Description** |
| --- | --- | --- | --- |
| Configuration memory ports | | | |
| from\_fpgacfg | in | t\_FROM\_FPGACFG; | Configuration registers bus |
| to\_tstcfg\_from\_rxtx | out | t\_TO\_TSTCFG\_FROM\_RXTX; |
| from\_tstcfg | in | t\_FROM\_TSTCFG; |
| TX path | | | |
| tx\_clk | in | 1 | TX interface clock |
| tx\_clk\_reset\_n | in | 1 | TX interface reset, active low |
| tx\_pct\_loss\_flg | out | 1 | TX packet loss flag, 0 - No packet loss, 1 - Packet lost. |
| tx\_txant\_en | out | 1 | TX transmit flag. 0 - No transmission, 1 - TX is transmitting samples. |
| TX interface data | | | |
| tx\_smpl\_fifo\_wrreq | out | TX\_IQ\_WIDTH | Write request |
| tx\_smpl\_fifo\_wrfull | in | 1 | Write full |
| tx\_smpl\_fifo\_wrusedw | in | TX\_SMPL\_FIFO\_WRUSEDW\_W | Write used words |
| tx\_smpl\_fifo\_data | out | 128 | Write data |
| TX FIFO read ports | | | |
| tx\_in\_pct\_reset\_n\_req | out | 1 | TX packet buffer reset request, active low |
| tx\_in\_pct\_rdreq | out | 1 | TX packet buffer read request |
| tx\_in\_pct\_data | in | TX\_IN\_PCT\_DATA\_W | TX packet buffer read data |
| tx\_in\_pct\_rdempty | in | 1 | TX packet buffer read empty |
| tx\_in\_pct\_rdusedw | in | TX\_IN\_PCT\_RDUSEDW\_W | TX packet buffer read used words |
| TX FIFO read ports | | | |
| tx\_in\_pct\_reset\_n\_req | out | 1 | TX packet FIFO reset request. |
| tx\_in\_pct\_rdreq | out | 1 | TX packet FIFO read request. |
| tx\_in\_pct\_data | in | TX\_IN\_PCT\_DATA\_W | TX packet FIFO read data request. |
| tx\_in\_pct\_rdempty | in | 1 | TX packet FIFO read empty. |
| tx\_in\_pct\_rdusedws | out | TX\_IN\_PCT\_RDUSEDW\_W | TX packet FIFO read used words. |
| RX path | | | |
| rx\_clk | in | 1 | RX interface clock |
| rx\_clk\_reset\_n | in | 1 | RX interface reset, active low |
| RX Sample FIFO ports | | | |
| rx\_smpl\_fifo\_wrreq | in | 1 | RX sample FIFO write request. |
| rx\_smpl\_fifo\_data | in | RX\_IQ\_WIDTH\*4 | RX sample FIFO write data. |
| rx\_smpl\_fifo\_wrfull | out | 1 | RX sample FIFO write full. |
| RX Packet FIFO ports | | | |
| rx\_pct\_fifo\_aclrn\_req | out | 1 | RX packet buffer reset request, active low |
| rx\_pct\_fifo\_wusedw | in | RX\_PCT\_BUFF\_WRUSEDW\_W | RX packet buffer write used words |
| rx\_pct\_fifo\_wrreq | out | 1 | RX packet buffer write request |
| rx\_pct\_fifo\_wdata | out | 64 | RX packet buffer write data |

### Receive interface – rx\_path\_top

Once rx\_path\_top **Figure 6** is enabled diq2fifo and data2packets modules starts continuously packing IQ samples into 4kB packets. For packet structure see [Stream protocol](https://github.com/myriadrf/LimeSuite/blob/master/docs/StreamProtocol.pdf) document.

Packets are written to 16kB F2H\_S0 FIFO buffer to maintain continuous data flow in short periods when PCIe host cannot accept data. If PCIe host halts data transfer for longer time period and four packets are buffered into 16kB buffer, FIFO full condition arises and other packets are dropped. When host starts to receive data after FIFO full condition, host should expect to receive those two four packets.

Module rx\_path\_top provides two 64bit sample counters. One is for TX logic – tx\_path\_top. TX logic uses this counter to synchronize transmitted LMS\_DQ1 samples with received LMS\_DIQ2 samples. Other is used for LMS\_DI2 samples packing into 4kB packets.

When rx\_path\_top is enabled diq2fifo module starts to collect IQ samples from LMS\_DIQ2 bus, collected samples are written to FIFO buffer and each write enables smpl\_cnt:inst4 module to increase its counter value. This means that counter value increases in same continuous rate as IQ sample rate.

Module smpl\_cnt:inst3 is used for LMS\_DIQ2 samples packing into 4kB packets. Module data2packets reads IQ samples in bursts from FIFO buffer, each read enables smpl\_cnt:inst3 module to increase its counter value. One read burst fills one 4kB packet and there are some idle cycles between bursts.



Figure 6 rx\_path\_top block diagram

Table 20 rx\_path\_top inctance description

| **Instance** | **Description** |
| --- | --- |
| fifo\_inst | FIFO buffer for storing samples. |
| data2packets | Module for packing IQ samples to 4kB packets. |
| smpl\_cnt:inst3 | Sample counter for tx\_path\_top. |
| smpl\_cnt:inst4 | Sample counter for data2packets module. |

### Transmit interface – tx\_path\_top

Transmit module tx\_path\_top reads IQ samples from H2F\_S0\_0 FIFO buffer packed in 4kB packets. Packet header (see [Stream protocol](https://github.com/myriadrf/LimeSuite/blob/master/docs/StreamProtocol.pdf) document) contains sample number (or so-called time stamp) at which packet should be transmitted.

By using sample numbers from rx\_path\_top and received sample numbers in packet header transmitted IQ samples can be synchronized with received IQ samples.

Module p2d\_wr\_fsm separates packet header and payload. Packet payload is written into one of four 4kB FIFO buffers located in packets2data module and packet header is stored in p2d\_rd module. This module can work in two modes:

* **Synchronization enabled** - module compares received sample number from packet header and sample number from rx\_path\_top. When sample number from received packet is equal to sample number of rx\_path\_top module (this means that it is time to send TX packet), read process begins and IQ samples are transmitted to LMS\_DIQ1 interface. When sample number from received packet is greater than sample number of rx\_path\_top module (this means that received packet should be sent after some time) p2d\_rd waits until those sample number will be equal. When sample number from received packet is less than sample number of rx\_path\_top module (this means that packet arrived too late) corresponding FIFO buffer is cleared.
* **Synchronization disabled** – module does not compare sample numbers and every received packet is transmitted to LMS\_DIQ1 interface.

Block diagram can be found in **Figure 7** and instance description in **Table 21**.



Figure 7 tx\_path\_top block diagram

Table 21 tx\_path\_top instance description

| **Instance** | **Description** |
| --- | --- |
| packets2data\_top | Wrapper file |
| packets2data | Wrapper file |
| p2d\_wr\_fsm | Module reads packets from EP01\_0 buffer and places to one of the 4kB FIFO buffers in increasing order and stores corresponding sample number from packet header. |
| p2d\_rd | Module checks one of the FIFO buffers if it is filled with samples in increasing order. When buffer is ready depending on received sample number from packet header and sample number from rx\_path\_top module buffer can be cleared or IQ sample reading begins. |
| fifo\_inst | FIFO buffer |
| sync\_fifo\_rw | Dual clock FIFO buffer for clock domain crossing. |

## Waveform player – wfm\_player\_x2\_top

Waveform player – wfm\_player\_x2\_top can be used to load waveform from H2F\_S0\_1, H2F\_S2\_1, H2F\_S2\_1, endpoint to external DDR3 memory and played back to LMS1\_DIQ1, LMS2\_DIQ1 and DAC5672 interface. Samples can be loaded using 4kB packets (see [Stream protocol](https://github.com/myriadrf/LimeSuite/blob/master/docs/StreamProtocol.pdf) document). External memory can store 512MB of data. Block diagram can be found in **Figure 8**.



Figure 8 wfm\_player\_x2\_top block diagram

Table 22 wfm\_player\_top instance description

| **Instance** | **Description** |
| --- | --- |
| wfm\_player | Waveform player instance, reads IQ packets from wfm\_0 or wfm\_1 FIFO buffer and writes to ddr3\_av\_2x32 module. |
| ddr3\_av\_2x32 | External DDR3 memory controller. |

Table 23 wfm\_player\_x2\_top parameters

| **Parameter** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- |
| DEV\_FAMILY | string | Cyclone V | Device family |
| External memory controller parameters | | | |
| mem\_cntrl\_rate | integer | 1 | External memory controller data rate setting |
| mem\_dq\_width | integer | 32 | External memory data width |
| mem\_dqs\_width | integer | 4 | External memory data strobe width |
| mem\_addr\_width | integer | 14 | External memory address width |
| mem\_ba\_width | integer | 3 | External memory bank address width |
| mem\_dm\_width | integer | 4 | External memory data mask width |
| Avalon 0 interface parameters | | | |
| avl\_0\_addr\_width | integer | 26 | Avalon bus address width |
| avl\_0\_data\_width | integer | 64 | Avalon bus data width |
| avl\_0\_burst\_count\_width | integer | 2 | Avalon bus burst count width |
| avl\_0\_be\_width | integer | 8 | Avalon bus byte enable width |
| avl\_0\_max\_burst\_count | integer | 2 | Avalon bus burst count width |
| avl\_0\_rd\_latency\_words | integer | 64 | Avalon bus latency words |
| Avalon 1 interface parameters | | | |
| avl\_1\_addr\_width | integer | 26 | Avalon bus address width |
| avl\_1\_data\_width | integer | 64 | Avalon bus data width |
| avl\_1\_burst\_count\_width | integer | 2 | Avalon bus burst count width |
| avl\_1\_be\_width | integer | 8 | Avalon bus byte enable width |
| avl\_1\_max\_burst\_count | integer | 2 | Avalon bus burst count width |
| avl\_1\_rd\_latency\_words | integer | 64 | Avalon bus latency words |
| wfm 0 player parameters | | | |
| wfm\_0\_infifo\_rdusedw\_width | integer | 11 | wfm\_0 INFIFO read used words width |
| wfm\_0\_infifo\_rdata\_width | integer | 64 | wfm\_0 INFIFO read data width |
| wfm\_0\_outfifo\_wrusedw\_width | integer | 9 | wfm\_0 OUTFIFO write used words width |
| wfm 1 player parameters | | | |
| wfm\_1\_infifo\_rdusedw\_width | integer | 11 | wfm\_0 INFIFO read used words width |
| wfm\_1\_infifo\_rdata\_width | integer | 64 | wfm\_0 INFIFO read data width |
| wfm\_1\_outfifo\_wrusedw\_width | integer | 9 | wfm\_0 OUTFIFO write used words width |

Table 24 wfm\_player\_x2\_top port description

| **Port** | **Type** | **Width** | **Description** |
| --- | --- | --- | --- |
| clk | in | 1 | Free running clock |
| reset\_n | in | 1 | Asynchronous, active low reset |
| WFM port 0 | | | |
| from\_fpgacfg\_0 | in | - | From configuration registers |
| wfm\_0\_infifo\_rdreq | out | 1 | Input FIFO read request |
| wfm\_0\_infifo\_rdata | in | 64 | Input FIFO read data |
| wfm\_0\_infifo\_rdempty | in | 1 | Input FIFO read empty |
| wfm\_0\_infifo\_rdusedw | in | 11 | Input FIFO read used words |
| wfm\_0\_outfifo\_reset\_n | out | 1 | Output FIFO reset request |
| wfm\_0\_outfifo\_wrreq | out | 1 | Output FIFO write request |
| wfm\_0\_outfifo\_data | out | 128 | Output FIFO write data |
| wfm\_0\_outfifo\_wrusedw | in | 9 | Output FIFO write used words |
| WFM port 1 | | | |
| from\_fpgacfg\_1 | in | - | From configuration registers |
| wfm\_1\_infifo\_rdreq | out | 1 | Input FIFO read request |
| wfm\_1\_infifo\_rdata | in | 64 | Input FIFO read data |
| wfm\_1\_infifo\_rdempty | in | 1 | Input FIFO read empty |
| wfm\_1\_infifo\_rdusedw | in | 11 | Input FIFO read used words |
| wfm\_1\_outfifo\_reset\_n | out | 1 | Output FIFO reset request |
| wfm\_1\_outfifo\_wrreq | out | 1 | Output FIFO write request |
| wfm\_1\_outfifo\_data | out | 128 | Output FIFO write data |
| wfm\_1\_outfifo\_wrusedw | in | 9 | Output FIFO write used words |
| External memory ports | | | |
| mem\_a | out | 14 | External memory pins |
| mem\_ba | out | 3 |
| mem\_ck | out | 1 |
| mem\_ck\_n | out | 1 |
| mem\_cke | out | 1 |
| mem\_cs\_n | out | 1 |
| mem\_dm | out | 4 |
| mem\_ras\_n | out | 1 |
| mem\_cas\_n | out | 1 |
| mem\_we\_n | out | 1 |
| mem\_reset\_n | out | 1 |
| mem\_dq | inout | 32 |
| mem\_dqs | inout | 4 |
| mem\_dqs\_n | inout | 4 |
| mem\_odt | out | 1 |
| phy\_clk | out | 1 |
| oct\_rzqin | in | 1 |

## LMS7002 interface – lms7002\_top

Module called lms7002\_top (see **Figure 9**) is used to send and receive data to/from LMS7002 IC. For transmit side IQ samples are written to regular FIFO interface and module then transfers data to LMS7002 physical DIQ1 interface. For receive side module collects IQ samples from physical LMS7002 DIQ2 interface and transforms them to simple data words with data valid signal. Instance decription can be found in **Table 25**.



Figure 9 Module lms7002\_top block diagram

Table 25 lms7002\_top instance description

| **Instance** | **Description** |
| --- | --- |
| diq2fifo | IQ data receive instance, collects samples from DIQ2. |
| lms7002\_tx | Transmits data from FIFO to DIQ1 interface. |

| **Parameter** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- |
| g\_DEV\_FAMILY | string | Cyclone V | Device family |
| g\_IQ\_WIDTH | integer | 12 | LMS7002 IQ bus width |
| g\_INV\_INPUT\_CLK | string | ON | DDIO cell clocking |
| g\_TX\_SMPL\_FIFO\_0\_WRUSEDW | integer | 9 | TX FIFO 0 buffer size |
| g\_TX\_SMPL\_FIFO\_0\_DATAW | integer | 128 | TX FIFO 0 buffer write data width |
| g\_TX\_SMPL\_FIFO\_1\_WRUSEDW | integer | 9 | TX FIFO 1 buffer size |
| g\_TX\_SMPL\_FIFO\_1\_DATAW | integer | 128 | TX FIFO 1 buffer write data width |
| Configurations bus | | | |
| from\_fpgacfg | integer | 12 | RX IQ sample width |
| from\_tstcfg | string | ON | Clock invert option on LMS\_DIQ2 interface |
| LMS7002 port 1 | | | |
| MCLK1 | in | 1 | Master clock for DIQ1 bus |
| FCLK1 | out | 1 | Feedback clock for DIQ1 bus |
| DIQ1 | out | 12 | TX IQ bus |
| ENABLE\_IQSEL1 | out | 1 |
| TXNRX1 | out | 1 | DIQ1 bus direction |
| LMS7002 port 2 | | | |
| MCLK2 | in | 1 | Master clock for DIQ2 bus |
| FCLK2 | out | 1 | Feedback clock for DIQ2 bus |
| DIQ2 | in | 11 | RX IQ bus |
| ENABLE\_IQSEL2 | in | 1 |
| TXNRX2 | out | 1 | DIQ1 bus direction |
| LMS7002 misc | | | |
| RESET | out | 1 | LMS7002 IC reset (active low) |
| TXEN | out | 1 | LMS7002 transmit enable |
| RXEN | out | 1 | LMS7002 receive enable |
| CORE\_LDO\_EN | out | 1 | External enable control signal for the internal LDO’s |
| Internal TX ports | | | |
| tx\_reset\_n | in | 1 | TX interface reset (active low) |
| tx\_fifo\_0\_wrclk | in | 1 | TX FIFO 0 write clock |
| tx\_fifo\_0\_reset\_n | in | 1 | TX FIFO 0 reset (active low) |
| tx\_fifo\_0\_wrreq | in | 1 | TX FIFO 0 write request |
| tx\_fifo\_0\_data | in | 128 | TX FIFO 0 write data |
| tx\_fifo\_0\_wrfull | out | 1 | TX FIFO 0 write full |
| tx\_fifo\_0\_wrusedw | out | 9 | TX FIFO 0 write used words |
| tx\_fifo\_1\_wrclk | in | 1 | TX FIFO 1 write clock |
| tx\_fifo\_1\_reset\_n | in | 1 | TX FIFO 1 reset (active low) |
| tx\_fifo\_1\_wrreq | in | 1 | TX FIFO 1 write request |
| tx\_fifo\_1\_data | in | 128 | TX FIFO 1 write data |
| tx\_fifo\_1\_wrfull | out | 1 | TX FIFO 1 write full |
| tx\_fifo\_1\_wrusedw | out | 9 | TX FIFO 1 write used words |
| Internal RX ports | | | |
| rx\_reset\_n | in | 1 | RX interface reset (active low) |
| rx\_diq\_h | out | 13 | RX IQ data from DDIO cell |
| rx\_diq\_l | out | 13 |
| rx\_data\_valid | out | 1 | RX data valid |
| rx\_data | out | 48 | RX data |
| RX sample compare | | | |
| rx\_smpl\_cmp\_start | in | 1 | Enable sample comparing with constant |
| rx\_smpl\_cmp\_length | in | 16 | Sample compare length in words |
| rx\_smpl\_cmp\_done | out | 1 | Sample compare done |
| rx\_smpl\_cmp\_err | out | 1 | Sample compare error |

## External ADC – adc\_top

External Analog to Digital converter module (see **Figure 10**) captures data from external ADS4246 IC. For instance description see **Table 26**.



Figure 10 Module adc\_top block diagram

Table 26 adc\_top instance description

| **Instance** | **Description** |
| --- | --- |
| ADS4246 | Data capture module |
| rx\_chain | Gain and IQ correctors for ADC IQ data |

Table 27 adc\_top parameter description

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Type** | **Default** | **Description** |
| dev\_family | string | CYCLONE V | FPGA device family name |
| data\_width | integer | 7 | ADC bus width |
| smpls\_to\_capture | integer | 4 | Number of samples to capture into output bus |

Table 28 adc\_top port description

| **Port** | **Type** | **Width** | **Description** |
| --- | --- | --- | --- |
| clk | in | 1 | Free running clock |
| reset\_n | in | 1 | Asynchronous, active low reset |
| ADC inputs | | | |
| ch\_a | in | 7 | Input to DDR cells from pins |
| ch\_b | in | 7 |
| SDR parallel output data | | | |
| data\_ch\_a | out | 14 | Captured samples channel A |
| data\_ch\_b | out | 14 | Captured samples channel B |
| Interleaved samples of both channels | | | |
| data\_ch\_ab | out | 1 | Captured ADC data |
| data\_ch\_ab\_valid | out | 2 | Captured ADC data valid |
| TX0 FIFO source for DAC | | | |
| to\_rxtspcfg | out | - | Configuration register bus |
| from\_rxtspcfg | in | - |

## External DAC – dac5672\_top

External Digital to Analog converter – dac5672 module (see **Figure 11**) transfers data written to regular FIFO to physical DAC5672 IC bus. For instance description see **Table 29.**



Figure 11 Module dac5672\_top block diagram

Table 29 dac5672\_top instance description

| **Instance** | **Description** |
| --- | --- |
| fifo\_inst | FIFO buffer |
| txiq\_par | Takes interleaved data stream and converts to parallel data stream |
| tx\_chain | Contains configurable gain, phase and IQ correctors and NCO |
| altddio\_out | Altera DDR output cell IP |

Table 30 dac5672\_top module parameters

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Type** | **Default** | **Description** |
| g\_DEV\_FAMILY | string | CYCLONE V GX | FPGA device family name |
| g\_IQ\_WIDTH | integer | 14 | DAC bus width |
| g\_TX0\_FIFO\_WRUSEDW | integer | 9 | TX0 FIFO write used words size |
| g\_TX0\_FIFO\_DATAW | integer | 128 | TX0 FIFO write data width |
| g\_TX1\_FIFO\_WRUSEDW | integer | 9 | TX1 FIFO write used words size |
| g\_TX1\_FIFO\_DATAW | integer | 128 | TX1 FIFO write data width |
| g\_TX2\_FIFO\_WRUSEDW | integer | 9 | TX2 FIFO write used words size |
| g\_TX2\_FIFO\_DATAW | integer | 128 | TX2 FIFO write data width |

Table 31 dac5672\_top module port description

| **Port** | **Type** | **Width** | **Description** |
| --- | --- | --- | --- |
| clk | in | 1 | Free running clock |
| reset\_n | in | 1 | Asynchronous, active low reset |
| DAC#1 Outputs | | | |
| DAC1\_SLEEP | out | 1 | DAC#1 mode selection |
| DAC1\_MODE | out | 1 |
| DAC1\_DA | out | 14 | DAC# 1 channel A data |
| DAC1\_DB | out | 14 | DAC# 1 channel B data |
| DAC#2 Outputs | | | |
| DAC1\_SLEEP | out | 1 | DAC#1 mode selection |
| DAC1\_MODE | out | 1 |
| DAC1\_DA | out | 14 | DAC# 1 channel A data |
| DAC1\_DB | out | 14 | DAC# 1 channel B data |
| Internal TX ports | | | |
| tx\_reset\_n | in | 1 | TX reset (active low) |
| tx\_src\_sel | in | 2 | TX source selection |
| TX0 FIFO source for DAC | | | |
| tx0\_wrclk | in | 1 | TX0 FIFO write clock |
| tx0\_reset\_n | in | 1 | TX0 FIFO reset (active low) |
| tx0\_wrfull | out | 1 | TX0 FIFO write full |
| tx0\_wrusedw | out | 9 | TX0 FIFO write used words |
| tx0\_wrreq | in | 1 | TX0 FIFO write request |
| tx0\_data | in | 128 | TX0 FIFO write data |
| TX1 FIFO source for DAC | | | |
| tx1\_wrclk | in | 1 | TX1 FIFO write clock |
| tx1\_reset\_n | in | 1 | TX1 FIFO reset (active low) |
| tx1\_wrfull | out | 1 | TX1 FIFO write full |
| tx1\_wrusedw | out | 9 | TX1 FIFO write used words |
| tx1\_wrreq | in | 1 | TX1 FIFO write request |
| tx1\_data | in | 128 | TX1 FIFO write data |
| TX2 FIFO source for DAC | | | |
| tx2\_wrclk | in | 1 | TX2 FIFO write clock |
| tx2\_reset\_n | in | 1 | TX2 FIFO reset (active low) |
| tx2\_wrfull | out | 1 | TX2 FIFO write full |
| tx2\_wrusedw | out | 9 | TX2 FIFO write used words |
| tx2\_wrreq | in | 1 | TX2 FIFO write request |
| tx2\_data | in | 128 | TX2 FIFO write data |
| TX2 source for DAC | | | |
| tx2\_dac1\_da | in | 14 | DAC# 1 channel A data source |
| tx2\_dac1\_db | in | 14 | DAC# 1 channel B data source |
| tx2\_dac2\_da | in | 14 | DAC# 2 channel A data source |
| tx2\_dac2\_db | in | 14 | DAC# 2 channel B data source |
| Configuration bus | | | |
| from\_fpgacfg | in | - | Configuration register bus |
| from\_txtspcfg\_0 | in | - |
| to\_txtspcfg\_0 | out | - |
| from\_txtspcfg\_1 | in | - |
| to\_txtspcfg\_1 | out | - |

## General periphery – general\_periph\_top

General periphery - general\_periph\_top module is responsible for controlling on board periphery such as LED, GPIO and Fan, default functions can be found in **Table 32**. Also default function can be overridden by internal registers see chapter **3.3 Softcore processor – nios\_cpu**.

Table 32 Default functions of LEDS, GPIO and fan

| **Schematic name** | **Board label** | **Type** | **Description** |
| --- | --- | --- | --- |
| FPGA\_LED1 | FPGA\_LED1 | Clock status | Blinking indicates presence of TCXO clock.  Colour indicates status of FPGA PLLs that are used for LMS digital interface clocking: Green – both PLLs are locked; Red/Green – at least one PLL is not locked. |
| FPGA\_LED2 | FPGA\_LED2 | TCXO control mode | No light – TCXO is controlled from DAC  Red – TCXO is controlled from phase detector and is not locked to external reference clock Green – TCXO is controlled from phase detector and is locked to external reference clock |
| FPGA\_LED3 | FPGA\_LED3 | RXPLL status | Indicates RXPLL lock status. 0 – no lock, 1 - locked |
| FPGA\_LED4 | FPGA\_LED4 | TXPLL status | Indicates TXPLL lock status. 0 – no lock, 1 - locked |
| FPGA\_LED5 | FPGA\_LED5 |  | - |
| FPGA\_LED6 | FPGA\_LED6 | - | - |
| FPGA\_GPIO0 | FPGA\_GPIO |  | Indicates when TX is transmitting IQ samples. 0 – not transmitting, 1 – transmitting. |
| FPGA\_GPIO1 |  | Indicates RXPLL lock status. 0 – no lock, 1 - locked |
| FPGA\_GPIO2 |  | Indicates TXPLL lock status. 0 – no lock, 1 - locked |
| FPGA\_GPIO3 |  | Indicates TX packet loss, 0 – no loss, 1 – packet lost. |
| FPGA\_GPIO4-15 |  | - |
| FAN\_CTRL | FAN |  | Fan control pin. Connected to LM75\_OS temperature sensor pin. |

Block diagram can be found in **Figure 12**, instances are described in **Table 33**. See **Table 34** and **Table 35** for module parameters and port description.



Figure 12 Module general\_periph\_top block diagram

Table 33 Module instance description

| **Instance** | **Description** |
| --- | --- |
| alive | Basic counter to implement blinking on led1. |
| FPGA\_LED1\_cntrl | Led1 control module, for showing clock status |
| FPGA\_LED2\_cntrl | Led2 control module, for showing TCXO control mode |
| onboard\_led | Led3-6 control module. |
| gpio\_ctrl | GPIO control instance |

Table 34 Module general\_periph\_top parameters

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Type** | **Default** | **Description** |
| DEV\_FAMILY | string | CYCLONE IV GX | FPGA device family name |
| N\_GPIO | integer | 16 | Number of GPIO used |

Table 35 Module general\_periph\_top input and output port description

| **Port** | **Type** | **Width** | **Description** |
| --- | --- | --- | --- |
| clk | in | 1 | Free running clock |
| reset\_n | in | 1 | Asynchronous, active low reset |
| Configuration bus | | | |
| from\_fpgacfg | in | - | Input/output ports from/to SPI configuration registers |
| to\_periphcfg | out | - |
| from\_periphcfg | in | - |
| LED1(Clock and PLL lock status) | | | |
| led1\_pll1\_locked | in | 1 | Lock status from PLL1 |
| led1\_pll2\_locked | in | 1 | Lock status from PLL2 |
| led1\_ctrl | in | 3 | led1\_ctrl[0]–manual LED control enable;led1\_ctrl[1]–red LED enable in manual mode;led1\_ctrl[2]–green LED enable in manual mode; |
| led1\_g | out | 1 | Output to dual color LED1 pin |
| led1\_r | out | 1 | Output to dual color LED1 pin |
| LED2(TCXO control status) | | | |
| led2\_clk | in | 1 | Clock from SPI master connected to DAC and ADF |
| led2\_adf\_muxout | in | 1 | Multiplexer output from ADF4002 |
| led2\_dac\_ss | in | 1 | DAC slave select |
| led2\_adf\_ss | in | 1 | ADF slave select |
| led2\_ctrl | in | 3 | led2\_ctrl[0]–manual LED control enable;led2\_ctrl[1]–red LED enable in manual mode;led2\_ctrl[2]–green LED enable in manual mode; |
| led2\_g | out | 1 | Output to dual color LED2 pin |
| led2\_r | out | 1 | Output to dual color LED2 pin |
| LED3-6 | | | |
| led3\_in | in | 1 | Input for controlling FPGA\_LED3 |
| led4\_in | in | 1 | Input for controlling FPGA\_LED4 |
| led5\_in | in | 1 | Input for controlling FPGA\_LED5 |
| led6\_in | in | 1 | Input for controlling FPGA\_LED6 |
| led3\_out | out | 1 | Output to FPGA\_LED3 pin |
| led4\_out | out | 1 | Output to FPGA\_LED4 pin |
| led5\_out | out | 1 | Output to FPGA\_LED5 pin |
| led6\_out | out | 1 | Output to FPGA\_LED6 pin |
| GPIO | | | |
| gpio\_dir | in | N\_GPIO | GPIO direction control, 0 – input, 1 – output |
| gpio\_out\_val | in | N\_GPIO | GPIO output value when direction is set to output |
| gpio\_rd\_val | out | N\_GPIO | GPIO input value vhen direction is set to input |
| gpio | inout | N\_GPIO | Connect to GPIO pins |
| Fan control | | | |
| fan\_sens\_in | in | 1 | From temperature sensor |
| fan\_ctrl\_out | out | 1 | To Fan control output |

## PLL module – pll\_top

PLL module – pll\_top (**Figure 13)** provides required clock sources for LM7002 RX and TX digital interfaces. Inside this module there are five dynamically reconfigurable PLL instances **Figure 14**. Clock frequency and phase relationship can be changed while FPGA is in user mode. Instance description can be found in **Table 36**.



Figure 13 PLL module – pll\_top



Figure 14 tx\_pll\_top\_cyc5/rx\_pll\_top\_cyc5 modules

Table 36. pll\_top module instance description

| **Instance** | **Description** |
| --- | --- |
| pll\_ps\_top | PLL dynamic phase reconfiguration module |
| adc\_dac\_pll | PLL dedicated for external ADC and DAC |
| tx\_pll\_top\_cyc5 | PLL dedicated for LMS7002 TX interface |
| rx\_pll\_top\_cyc5 | PLL dedicated for LMS7002 RX interface |

Table 37. pll\_top module parameters

| **Parameter** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- |
| INTENDED\_DEVICE\_FAMILY | string | Cyclone V |  |
| N\_PLL | integer | 5 | Number of PLLs |
| LMS#1 TX PLL parameters | | | |
| LMS1\_TXPLL\_DRCT\_C0\_NDLY | integer | 1 | Number of logic cells in clock output path |
| LMS1\_TXPLL\_DRCT\_C1\_NDLY | integer | 2 |
| LMS#1 RX PLL parameters | | | |
| LMS1\_RXPLL\_DRCT\_C0\_NDLY | integer | 1 | Number of logic cells in clock output path |
| LMS1\_RXPLL\_DRCT\_C1\_NDLY | integer | 2 |
| LMS#2 TX PLL parameters | | | |
| LMS2\_TXPLL\_DRCT\_C0\_NDLY | integer | 1 | Number of logic cells in clock output path |
| LMS2\_TXPLL\_DRCT\_C1\_NDLY | integer | 2 |
| LMS#2 RX PLL parameters | | | |
| LMS2\_RXPLL\_DRCT\_C0\_NDLY | integer | 1 | Number of logic cells in clock |
| LMS2\_RXPLL\_DRCT\_C1\_NDLY | integer | 2 | output path |

Table 38 pll\_top port description

| **Port** | **Type** | **Width** | **Description** |
| --- | --- | --- | --- |
| LMS#1 TX PLL ports | | | |
| lms1\_txpll\_inclk | in | 1 | PLL input clock from LMS\_MCLK1 pin |
| lms1\_txpll\_reconfig\_clk | in | 1 | Free running clock, used for PLL reconfiguration. |
| lms1\_txpll\_rcnfg\_to\_pll | in | 1 | PLL reconfiguration ports |
| lms1\_txpll\_rcnfg\_from\_pll | out | 1 | PLL reconfiguration ports |
| lms1\_txpll\_logic\_reset\_n | in | 1 | TX PLL logic reset, active low |
| lms1\_txpll\_clk\_ena | in | 1 | Clock enable for clock outputs |
| lms1\_txpll\_drct\_clk\_en | in | 1 | Direct clock enable to bypass PLL. |
| lms1\_txpll\_c0 | out | 1 | TX PLL c0 output clock |
| lms1\_txpll\_c1 | out | 1 | TX PLL c1 output clock (phase shifted version of c0) |
| lms1\_txpll\_locked | out | 1 | TX PLL lock status. Outputs high level when PLL is locked |
| LMS#1 RX PLL ports | | | |
| lms1\_rxpll\_inclk | in | 1 | PLL input clock from LMS1\_MCLK1 pin |
| lms1\_rxpll\_reconfig\_clk | in | 1 | Free running clock, used for PLL reconfiguration. |
| lms1\_rxpll\_rcnfg\_to\_pll | in | 1 | PLL reconfiguration ports |
| lms1\_rxpll\_rcnfg\_from\_pll | out | 1 | PLL reconfiguration ports |
| lms1\_rxpll\_logic\_reset\_n | in | 1 | TX PLL logic reset, active low |
| lms1\_rxpll\_clk\_ena | in | 1 | Clock enable for clock outputs |
| lms1\_rxpll\_drct\_clk\_en | in | 1 | Direct clock enable to bypass PLL. |
| lms1\_rxpll\_c0 | out | 1 | RX PLL c0 output clock |
| lms1\_rxpll\_c1 | out | 1 | RX PLL c1 output clock (phase shifted version of c0) |
| lms1\_rxpll\_locked | out | 1 | RX PLL lock status. Outputs high level when PLL is locked |
| Sample comparing ports from LMS#1 interface | | | |
| lms1\_smpl\_cmp\_en | out | 1 | Sample compare enable |
| lms1\_smpl\_cmp\_done | in | 1 | Sample compare done |
| lms1\_smpl\_cmp\_error | in | 1 | Sample compare error |
| lms1\_smpl\_cmp\_cnt | out | 16 | Sample compare word count |
| LMS#2 TX PLL ports | | | |
| lms2\_txpll\_inclk | in | 1 | PLL input clock from LMS\_MCLK1 pin |
| lms2\_txpll\_reconfig\_clk | in | 1 | Free running clock, used for PLL reconfiguration. |
| lms2\_txpll\_rcnfg\_to\_pll | in | 1 | PLL reconfiguration ports |
| lms2\_txpll\_rcnfg\_from\_pll | out | 1 | PLL reconfiguration ports |
| lms2\_txpll\_logic\_reset\_n | in | 1 | TX PLL logic reset, active low |
| lms2\_txpll\_clk\_ena | in | 1 | Clock enable for clock outputs |
| lms2\_txpll\_drct\_clk\_en | in | 1 | Direct clock enable to bypass PLL. |
| lms2\_txpll\_c0 | out | 1 | TX PLL c0 output clock |
| lms2\_txpll\_c1 | out | 1 | TX PLL c1 output clock (phase shifted version of c0) |
| lms2\_txpll\_locked | out | 1 | TX PLL lock status. Outputs high level when PLL is locked |
| LMS#2 RX PLL ports | | | |
| lms2\_rxpll\_inclk | in | 1 | PLL input clock from LMS1\_MCLK1 pin |
| lms2\_rxpll\_reconfig\_clk | in | 1 | Free running clock, used for PLL reconfiguration. |
| lms2\_rxpll\_rcnfg\_to\_pll | in | 1 | PLL reconfiguration ports |
| lms2\_rxpll\_rcnfg\_from\_pll | out | 1 | PLL reconfiguration ports |
| lms2\_rxpll\_logic\_reset\_n | in | 1 | TX PLL logic reset, active low |
| lms2\_rxpll\_clk\_ena | in | 1 | Clock enable for clock outputs |
| lms2\_rxpll\_drct\_clk\_en | in | 1 | Direct clock enable to bypass PLL. |
| lms2\_rxpll\_c0 | out | 1 | RX PLL c0 output clock |
| lms2\_rxpll\_c1 | out | 1 | RX PLL c1 output clock (phase shifted version of c0) |
| lms2\_rxpll\_locked | out | 1 | RX PLL lock status. Outputs high level when PLL is locked |
| Sample comparing ports from LMS#1 interface | | | |
| lms1\_smpl\_cmp\_en | out | 1 | Sample compare enable |
| lms1\_smpl\_cmp\_done | in | 1 | Sample compare done |
| lms1\_smpl\_cmp\_error | in | 1 | Sample compare error |
| lms1\_smpl\_cmp\_cnt | out | 16 | Sample compare word count |
| PLL for DAC, ADC | | | |
| pll\_0\_inclk | in | 1 | PLL input clock |
| pll\_0\_reconfig\_clk | in | 1 | Free running clock, used for PLL reconfiguration. |
| pll\_0\_rcnfg\_to\_pll | in | 1 | PLL reconfiguration ports |
| pll\_0\_rcnfg\_from\_pll | out | 1 | PLL reconfiguration ports |
| pll\_0\_logic\_reset\_n | in | 1 | PLL logic reset, active low |
| pll\_0\_clk\_ena | in | 1 | Clock enable for clock outputs |
| pll\_0\_drct\_clk\_en | out | 1 | Direct clock enable to bypass PLL. |
| pll\_0\_c0 | out | 1 | PLL c0 output clock |
| pll\_0\_c0\_pin | out | 1 | PLL c0 output clock, connected to output pin |
| pll\_0\_c1 | out | 1 | PLL c1 output clock (phase shifted version of c0) |
| pll\_0\_c1\_pin | out | 1 | PLL c1 output clock, connected to output pin |
| pll\_0\_locked | out | 1 | PLL lock status. Outputs high level when PLL is locked |
| Reconfiguration 0 ports | | | |
| rcnfg\_0\_mgmt\_readdata | in | 1 | Avalon reconfiguration bus from dynamic phase reconfiguration module (LMS#1). |
| rcnfg\_0\_mgmt\_waitrequest | in | 1 |
| rcnfg\_0\_mgmt\_read | out | 1 |
| rcnfg\_0\_mgmt\_write | out | 1 |
| rcnfg\_0\_mgmt\_address | out | 9 |
| rcnfg\_0\_mgmt\_writedata | out | 32 |
| Reconfiguration 1 ports | | | |
| rcnfg\_1\_mgmt\_readdata | in | 1 | Avalon reconfiguration bus from dynamic phase reconfiguration module (LMS#2). |
| rcnfg\_1\_mgmt\_waitrequest | in | 1 |
| rcnfg\_1\_mgmt\_read | out | 1 |
| rcnfg\_1\_mgmt\_write | out | 1 |
| rcnfg\_1\_mgmt\_address | out | 9 |
| rcnfg\_1\_mgmt\_writedata | out | 32 |
| Configuration registers | | | |
| to\_pllcfg | out | - | Input/output ports from/to SPI configuration registers |
| from\_pllcfg | in | - | Input/output ports from/to SPI configuration registers |

# Examples

In this chapter various examples can be found on how to use gateware.

## Accessing FPGA registers

Internal FPGA registers can be accessed using PCIe host via pipe-like device files. For Linux host they are named /dev/xillybus\_control0\_read\_32 and /dev/xillybus\_control0\_write\_32. For windows host they are named \\.\xillybus\_control0\_read\_32 and \\.\xillybus\_control0\_write\_32. See <http://xillybus.com/doc> for documentation. See **LMS64C\_protocol** document for protocol structure and description of commands used in examples. See chapter **3.3 Softcore processor – nios\_cpu** for internal FPGA register description.

**Read** – 64byte packet containing request command “CMD\_BRDSPI16\_RD” has to be sent from host to FPGA and 64 bytes response packet has to be read from FPGA to host. Read example reads 0x0000 address Board\_ID register value, which is 0x000F for LimeSDR-PCIe board.

Request – host writes 64B to /dev/xillybus\_control0\_write\_32 or \\.\xillybus\_control0\_write\_32:

Address

0000 56 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00

0010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Response –host reads 64B from /dev/xillybus\_control0\_read\_32 or \\.\xillybus\_control0\_read\_32:

Address

0000 56 01 01 00 00 00 00 00 00 00 00 0E 00 00 00 00

0010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

**Write** – 64byte packet containing request command “CMD\_BRDSPI16\_WR” has to be sent from host to FPGA and 64 bytes response packet has to be read from FPGA to host. Write example writes 0x1234 value to 0x00DF address. This register is currently reserved and has no dedicated function.

Request – host writes 64B to /dev/xillybus\_control0\_write\_32 or \\.\xillybus\_control0\_write\_32:

Address

0000 55 00 01 00 00 00 00 00 00 DF 12 34 00 00 00 00

0010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Response – host reads 64B from /dev/xillybus\_control0\_read\_32 or \\.\xillybus\_control0\_read\_32:

Address

0000 55 01 01 00 00 00 00 00 00 00 00 00 00 00 00 00

0010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

## Accessing LMS7002M registers

Configuration memory which is inside LMS7002M can be accessed using PCIe host via pipe-like device files. For Linux host they are named /dev/xillybus\_control0\_read\_32 and /dev/xillybus\_control0\_write\_32. For windows host they are named \\.\xillybus\_control0\_read\_32 and \\.\xillybus\_control0\_write\_32. See <http://xillybus.com/doc> for documentation. See **LMS64C\_protocol** document for protocol structure and description of commands used in examples. Registers map of LMS7002M can be found in [LMS7002M – Multi-Band, Multi-Standard MIMO, Programming and Calibration Guide](https://github.com/myriadrf/LMS7002M-docs/blob/master/LMS7002M_Programming_and_Calibration_Guide_v31r05.pdf).

**Read** – 64byte packet containing request command “CMD\_LMS7002\_RD” has to be sent from host to FPGA and 64 bytes response packet has to be read from FPGA to host. Read example reads 0x0020 address register value, which is 0xFFFF by default.

Request – host writes 64B to /dev/xillybus\_control0\_write\_32 or \\.\xillybus\_control0\_write\_32:

Address

0000 22 00 01 00 00 00 00 00 00 20 00 00 00 00 00 00

0010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Response – host reads 64B from /dev/xillybus\_control0\_read\_32 or \\.\xillybus\_control0\_read\_32:

Address

0000 22 01 01 00 00 00 00 00 00 00 FF FF 00 00 00 00

0010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

**Write** – 64byte packet containing request command “CMD\_LMS7002\_WR” has to be sent from host to FPGA and 64 bytes response packet has to be read from FPGA to host. Write example writes 0xE4E4 value to 0x0024 address.

Request – host writes 64B to /dev/xillybus\_control0\_write\_32 or \\.\xillybus\_control0\_write\_32: :

Address

0000 21 00 01 00 00 00 00 00 00 24 E4 E4 00 00 00 00

0010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Response – host reads 64B from /dev/xillybus\_control0\_read\_32 or \\.\xillybus\_control0\_read\_32:

Address

0000 21 01 01 00 00 00 00 00 00 00 00 00 00 00 00 00

0010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

## Periphery control

**LED control -** modify FPGA register as showed in **Table 39** to turn on and change colour of FPGA\_LED2.

Table 39 FPGA\_LED2 control example

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N** | **CMD** | **Address (HEX)** | **Value (HEX)** | **Description** |
| 1 | WR | 001A | 0010 | Override FPGA\_LED2 control |
| 2 | WR | 001A | 0030 | Turn on FPGA\_LED2\_R (red is on, green - off) |
| 3 | WR | 001A | 0050 | Turn on FPGA\_LED2\_G (green is on, red - off) |

## Configuring FPGA PLL module

To configure PLLs of pll\_top module LMS7002M chip has to be already configured and valid clock sources provided to LMS\_MCLK1 (connected to txpll\_top module) and LMS\_MCLK2 (connected to rxpll\_top module) pins. For LMS7002M chip configuration see chapter 4.2 Accessing LMS7002M registers.

Configuration of pll\_top module can be done by accessing FPGA registers see chapter **4.1 Accessing FPGA registers**. For register description see chapter **3.3 Softcore processor – nios\_cpu**.

PLL output frequency Fout can be calculated using following equation:

(1); (2); (3);

where *Fref*- PLL reference frequency, *FVCO* – VCO frequency, *FOUT* – Output frequency. See Cyclone IV datasheet for allowed frequency ranges.

### RX PLL module - rxpll\_top configuration (auto phase shift mode)

This example assumes that LMS7002M chip is already configured, outputs 20MHz clock on LMS\_MCLK2 pin and LMS\_DIQ2 interface outputs constant IQ values (I=0xAAA, Q=0x555). See **Table 40** for configuration sequence.

Table 40 rxpll\_top configuration sequence in auto phase shift mode

| **N** | **CMD** | **Address (HEX)** | **Value (HEX)** | **Description** |
| --- | --- | --- | --- | --- |
| 1 | WR | 0005 | 0000 | Turn off direct clocking |
| 2 | WR | 0025 | 01F0 | Set PLL parameters |
| 0023 | 0008 | Set PLL index to 1 and rest bits to zero |
| 3 | WR | 0023 | 0008 | Set PLL index to 1 and rest bits to zero |
| 0026 | 000A | N, M division bypass and odd division values. N, M division is not bypassed, odd division values enabled |
| 002A | 0201 | N, count value = 0x02 + 0x01 = 0x03 (3 DEC) |
| 002B | 6261 | M count value = 0x62 + 0x61 = 0xC3 (195 DEC) |
| 002E | 2120 | C0 count value = 0x21 + 0x20 = 0x41 (65 DEC) |
| 002F | 2120 | C1 count value = 0x21 + 0x20 = 0x41 (65 DEC) |
| 0027 | 555a | Counter C0-C7 bypass and odd division control bits. C0 and C1 not bypassed, others bypassed. C0 and C1 odd division values enabled, others not enabled. |
| 0028 | 5555 | Counter C7-C15 bypass and odd division control bits. All counters are bypassed. |
| 0023 | 0009 | Trigger reconfiguration for PLL index 1. |
| 4 | WR | 0023 | 6308 | Release PLL reconfiguration bit, set PLL index - 1, cnt index - 3, phase shift - up, phase shift mode - auto |
| 0024 | 0207 | Phase shift value = 0x0207 (519 DEC), represents 360 degrees (range in which auto phase shift is executed) |
| 0023 | 630a | Trigger auto phase shift for PLL index 1, cnt index 3, phase shift - up, phase shift mode - auto |
| 5 | RD | 0021 | - | Read PLL configuration status register and wait for configuration done (0x0005) |
| 6 | WR | 0023 | 6308 | Release PLL phase shift bit, set PLL index - 1, cnt index - 3, phase shift - up, phase shift mode - auto |

### TX PLL module - txpll\_top configuration (auto phase shift mode)

This example assumes that LMS7002M chip is already configured, outputs 20MHz clock on LMS\_MCLK1-2 pins, LimeLight digital loopback is enabled and FPGA rxpll\_top module is already configured. See **Table 41** for configuration sequence.

Table 41 txpll\_top configuration sequence in auto phase shift mode

| **N** | **CMD** | **Address (HEX)** | **Value (HEX)** | **Description** |
| --- | --- | --- | --- | --- |
| 1 | WR | 000A | 0200 | Enable TX test pattern |
| 2 | WR | 0005 | 0000 | Turn off direct clocking |
| 3 | WR | 0025 | 01F0 | Set PLL parameters |
| 0023 | 0000 | Set PLL index to 0 and rest bits to zero |
| 4 | WR | 0023 | 0000 | Set PLL index to 0 and rest bits to zero |
| 0026 | 000A | N, M division bypass and odd division values. N, M division is not bypassed, odd division values enabled |
| 0002A | 0201 | N, count value = 0x02 + 0x01 = 0x03 (3 DEC) |
| 002B | 6261 | M count value = 0x62 + 0x61 = 0xC3 (195 DEC) |
| 002E | 2120 | C0 count value = 0x21 + 0x20 = 0x41 (65 DEC) |
| 002F | 2120 | C1 count value = 0x21 + 0x20 = 0x41 (65 DEC) |
| 0027 | 555a | Counter C0-C7 bypass and odd division control bits. C0 and C1 not bypassed, others bypassed. C0 and C1 odd division values enabled, others not enabled. |
| 0028 | 5555 | Counter C7-C15 bypass and odd division control bits. All counters are bypassed. |
| 0023 | 0001 | Trigger reconfiguration for PLL index 0. |
| 5 | WR | 0023 | 6300 | Release PLL reconfiguration bit, set PLL index - 0, cnt index - 3, phase shift - up, phase shift mode - auto |
| 0024 | 0207 | Phase shift value = 0x0207 (519 DEC), represents 360 degrees (range in which auto phase shift is executed) |
| 0023 | 6302 | Trigger auto phase shift for PLL index 0, cnt index 3, phase shift - up, phase shift mode - auto |
| 6 | RD | 0021 | - | Read PLL configuration status register and wait for configuration done (0x0005) |
| 7 | WR | 0023 | 6300 | Release PLL phase shift bit, set PLL index - 0, cnt index - 3, phase shift - up, phase shift mode - auto |

## Controlling TX and RX data stream

Data stream can be enabled when LMS7002M chip and FPGA PLL modules are configured. See chapters **4.2 Accessing LMS7002M registers** and **4.4 Configuring FPGA PLL module.**

**To enable TX and RX data stream –** follow FPGA register write sequence described in **Table 42**.

Table 42 enabling TX and RX data stream

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N** | **CMD** | **Address (HEX)** | **Value (HEX)** | **Description** |
| 1 | WR | 000A | 0000 | Stop data stream |
| 2 | WR | 0009 | 0000 | Clear packet loss and reset timestamp bits. |
| 3 | WR | 0009 | 0003 | Clear packet loss flag and reset timestamp. |
| 4 | WR | 0009 | 0000 | Clear packet loss and reset timestamp bits. |
| 5 |  |  |  | Reset xillybus\_write\_32 and xillybus\_read\_32 streams |
| 6 | WR | 0008 | 102 | Set sample width -12, mode - TRXIQ, DDR - enabled, TRXIQ\_PULSE mode - disabled, packet synchronization - enabled |
| 7 | WR | 0007 | 0001 | Set active channels - 1 |
| 8 | WR | 000A | 0001 | Start stream |

**To disable** **TX and RX data stream –** follow FPGA register write sequence described in **Table 43.**

Table 43 disabling TX and RX data stream

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N** | **CMD** | **Address (HEX)** | **Value (HEX)** | **Description** |
| 1 | WR | 000A | 0000 | Stop data stream |

## Using WFM player

WFM player requires that LMS7002M has to be configured. See **Table 44** for data loading sequence.

Table 44 WFM data loading

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N** | **CMD** | **Address (HEX)** | **Value (HEX)** | **Description** |
| 1 | WR | 000C | 0003 | Enable both channels |
| 2 | WR | 000E | 0002 | Set sample width to 16bit mode |
| 4 | WR | 000D | 0006 | Enable WFM loading |
| 5 |  |  |  | Load WFM data to xillybus\_write\_32 |
| 6 | WR | 000D | 0002 | Disable WFM loading, start playing file |