Contents

© Copyright 2016 Xilinx

Preface

This document details The JESD204B TI reference design for the TI TSW14J01 or Xilinx VC707 / KC705 / ZC706 evaluation kit's using TI's TSW14J10 interposer.

Important

This reference design should not be used as an example for how to connect a specific ADC or DAC to a Xilinx FPGA. This reference design is capable of interfacing with all TI the FMC based ADC and DAC EVM's. To do this the design complexity is much greater than is required to simply interface to an ADC or DAC running with a specific configuration and much of the design has been abstracted and placed under HSDC Pro software control.

For a quick start to get your own design up and running. Refer to [1] Xilinx JESD204B LogiCORE IP Product Guide and start with the example design and demo testbench that is delivered when you generate a customized JESD204B core.

© Copyright 2016 Xilinx

JESD204B TI Reference design

The TSW14J01 or VC707 / KC705 / ZC706 plus TSW14J10 interposer are used as evaluation platforms for TI ADC and DAC products interfacing to Xilinx FPGA's and Xilinx JESD204B Intellectual property (IP). All platforms work in conjunction with TI ADC and DAC EVM's and TI's PC based convertor evaluation software High Speed Data Convertor (HSDC) Pro. Figure's 1 shows an example of the system.

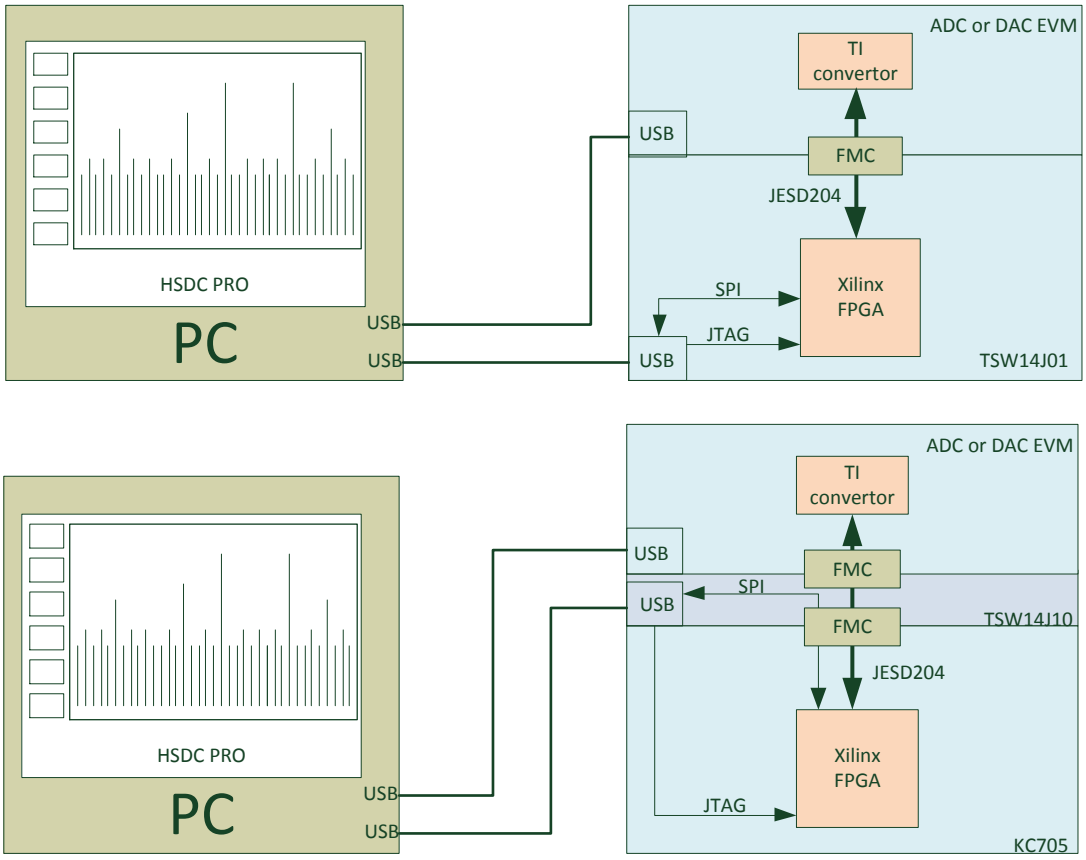


Figure 1 TSW14J01 and VC707/KC705/ZC706 plus TSW14J10 convertor evaluation system using HSDC Pro

This document assumes the user is familiar with the operation and functionality of HSDC Pro for convertor evaluation. Please see [1] for details. The HSDC Pro installer includes pre built bitstreams generated using this reference design. You do not need to build this project to use HSDC Pro.

References

[1] SLWU087A.pdf HSDC Pro GUI User Guide available from www.ti.com.

[2] LogiCORE IP JESD204B v7.0 "pg066-jesd204.pdf".

[3] 7 Series FPGAs GTX/GTH Transceivers user guide "ug476_7Series_Transceivers.pdf".

[4] Kintex®-7 FPGAs Data Sheet: DC and Switching Characteristics "ds182_Kintex_7_Data_Sheet.pdf"

© Copyright 2016 Xilinx

Overview

Figure 2 shows a high level block diagram of this design.

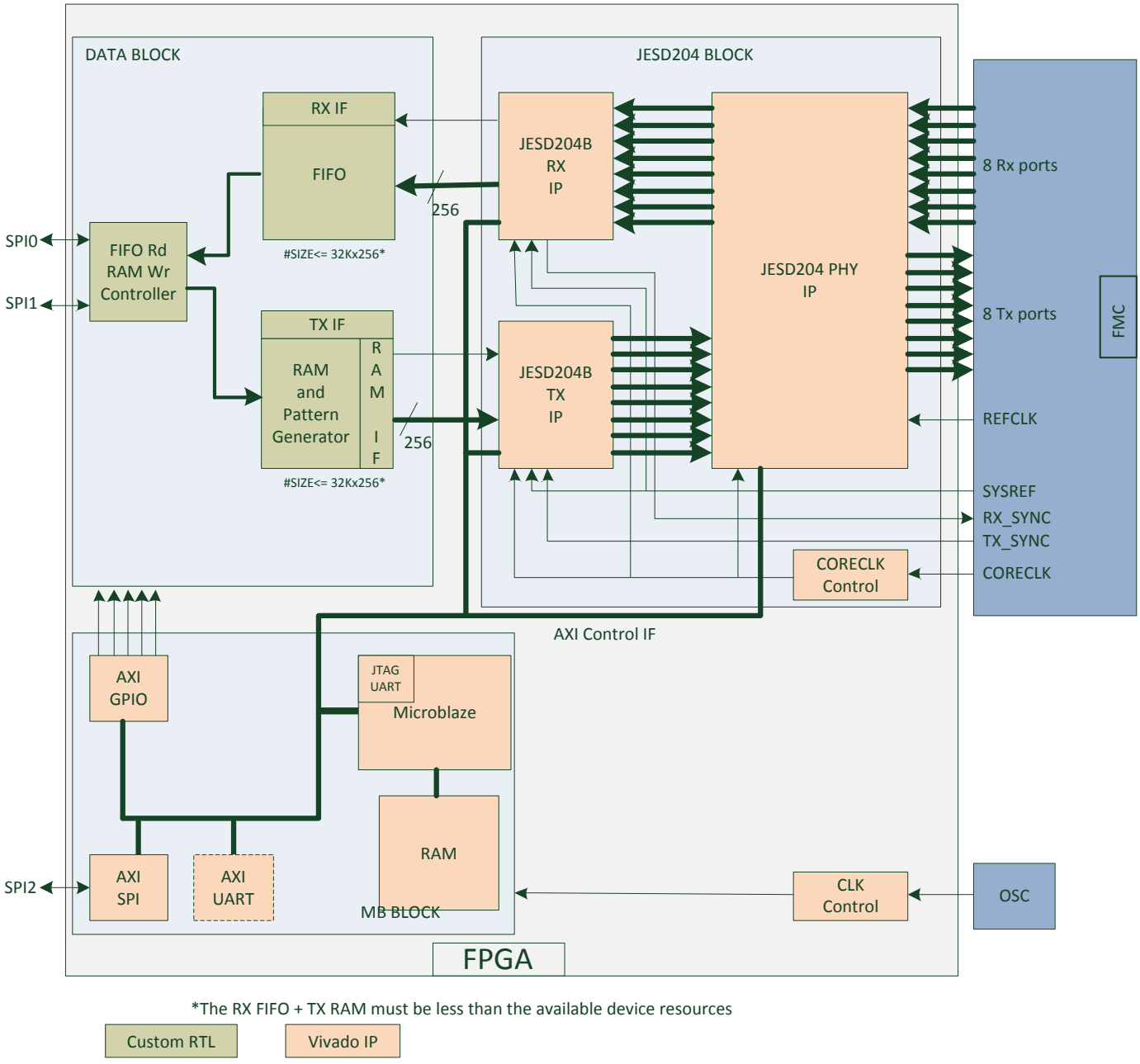


Figure 2 FPGA Block Diagram

© Copyright 2016 Xilinx

This reference design shows the following:

- Transceiver sharing between TX and RX JESD204B cores using the JESD204B PHY core.
- Line rate switching through the full supported frequency range of the transceivers under software control.
- Interfacing to TX and RX JESD204B cores for data ingress and egress.
- Interfacing to TX and RX JESD204B cores for configuration of JESD204B link parameters using a MicroBlaze processor.
- Packaging hdl and IP blocks for use in IP integrator using TCL.
- Creating a JESD204B block diagram based design in IP integrator using TCL.

How to build

The reference design is built by running a single script as follows:

1. Ensure you have Vivado 2016.1 installed.
2. Unzip the reference design into a folder of your choice.
3. Open the 2016.1 Vivado GUI and change directory into the unzipped reference design folder. You can change directory at the tcl prompt by simply typing ***cd <path/dir>***. Also the command ***pwd*** will return the current working directory.
4. Type ***source ./script/build_it.tcl*** at the Vivado TCL console.
5. Wait for the build to complete.

The reference design is configured by default to build for the VC707 platform. If you wish to build for the KC705 / ZC706 or TSW14J01. Simply edit the BOARD variable at the start of the build_it.tcl script.

Please see the scripts section of this document for a detailed overview of the operations performed by this script and the sub-scripts called by it.

The build it script will create a project ***proj_VC707*** and build it. Upon completion the implemented design will be opened.

After building the FPGA project you may rebuild the MicroBlaze software ELF file as follows:

1. Ensure you have built the project using the steps outlined above.
2. Export the hardware from Vivado and launch the SDK.
3. Using the SDK. Create a new project of type **Xilinx Application Project.** Name this project ***tsw*** and create it based on the Empty Application Template.
4. Copy .c and .h files from folder ***./sw_src/tsw/src*** to ./***proj_VC707/proj_VC707.sdk/tsw/src***
5. Refresh the project view (press F5 or choose refresh from the file menu). The copied project files should then be visible and compile cleanly.

© Copyright 2016 Xilinx

Design

Interfaces

The following physical ports are included in the design.

- **JESD204B:** There are 16 JESD204B lanes. One 8 lane receive link and one 8 lane transmit link. These ports are connected to the FMC HPC connector along with REFCLK and GLBLCLK to allow the connection of an ADC or DAC EVM.
- **SPI Control Port:** SPI port 2 is used as the control interface for the FPGA. Commands are written from the host PC to configure and control the programmable parameters of JESD204B sub-system. This SPI port uses the Xilinx SPI slave IP and is configured for 32 bit data transfers.
- **SPI DATA Ports**: SPI ports 0 and 1 are dedicated to transferring sample data to and from the host PC. These ports are only used to read from the data_block RX FIFO's and write to the data_block TX RAM's. The ports are configured for 32 bit data words.
- **UART**: There is a UART in the MicroBlaze sub-system. This UART is used to output system debug information. This port is connected to the USB UART on the VC707 and KC705. On the ZC706 this port is connected to a JTAG UART in the MicroBlaze Debug Module.
- **LED's**: There is a bank of LED's on each board used to show status. Note the polarity of the LED's on the TSW14J01 is reversed.
- **RESET Push button:** A soft reset of the system can be performed from a push button. This is connected to the **NORTH SW3** button on the VC707 and **NORTH SW2** on the KC705.

Programmable Parameters

These parameters are changeable under software control from HSDC Pro using the USB to SPI interface to the FPGA.

- JESD204B GTX REFCLK and line rate (see appendix 2 for an explanation of the parameters being reprogrammed).
- The JESD204B lanes in use (see [1] for a description on the lanes in use register).
  1 to 8 active transmitter ports.
  1 to 8 active receiver ports.
- JESD204B TX and RX link parameters.
  As specified in [1] page 99 Table 20 "Link configuration parameters".

See Appendix 1 for a full map and description of the registers accessible via the SPI control interface.

© Copyright 2016 Xilinx

MicroBlaze Block

The **mb_block** block implements the MicroBlaze processor susbsystem. Figure 3 shows an overview the internals of the **mb_block** (please refer to the mb_block created within IP Integrator for a more detailed view).

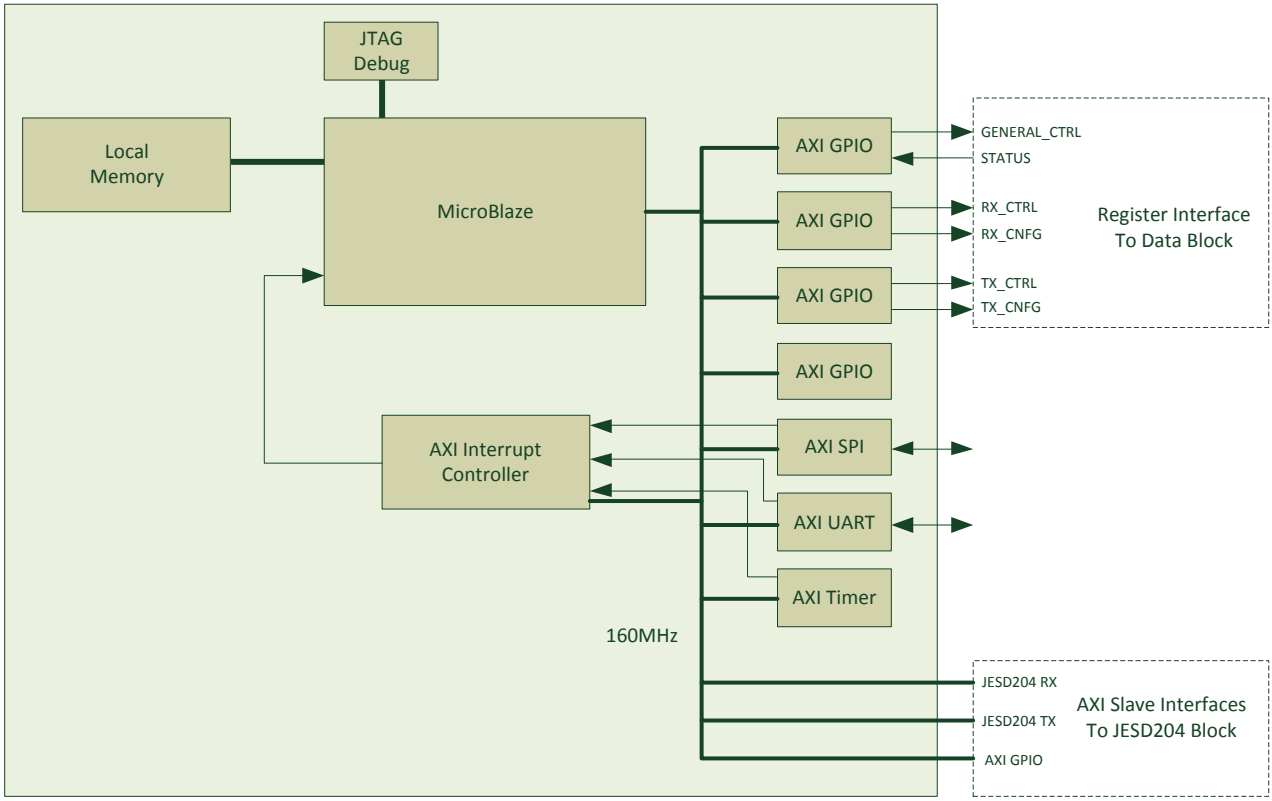


Figure 3 mb_block

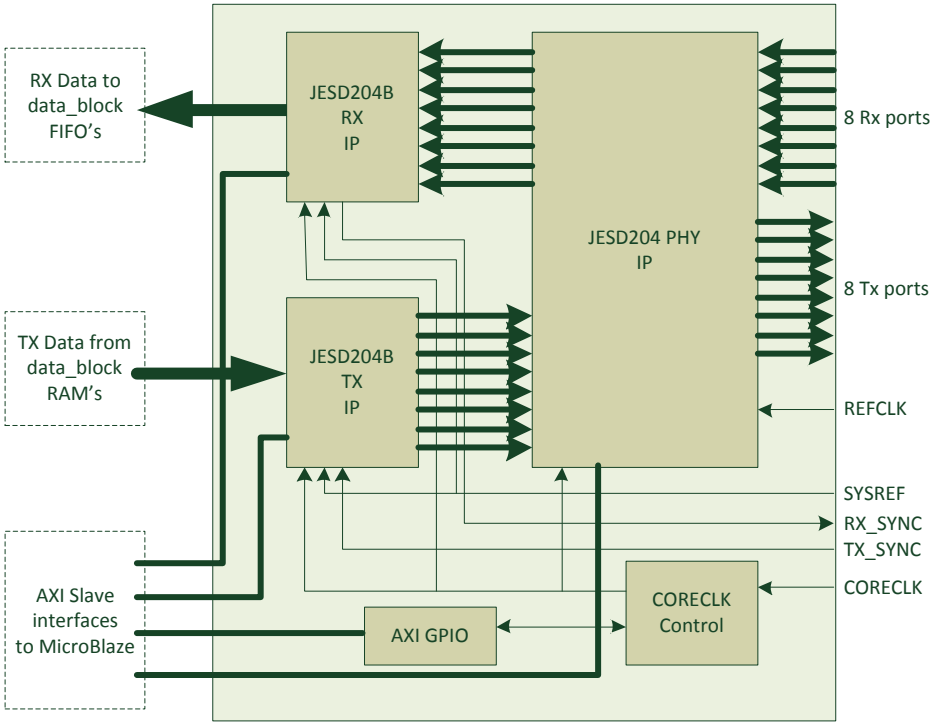
The MicroBlaze processor block is configured as follows:

- A MicroBlaze Debug Monitor provides JTAG access for debug and software development.
- Internal FPGA Block RAM's are used for program and data memory. The MicroBlaze does not use any off chip memory.
- An AXI SPI slave interface provides the control interface to the system. This SPI interface is both an SPI slave and an AXI slave. This interface is configured to generate an interrupt when its receive buffer is full.
- An AXI UART interface. This interface is provided purely for debug.
- Multiple AXI GPIO interfaces are setup to provide the register interface to the **data_block**
- An AXI GPIO to drive the on board LED's (Note the LED's must be connected to the **mb_block** for this to function. By default the LED's are connected to the jesd204_block)
- Three AXI interface ports for connection to the jesd204_block.

© Copyright 2016 Xilinx

JESD204 Block

The **jesd204_block** implements the JESD204B TX, RX and PHY IP and subsystem. Figure 4 shows an overview the internals of the **jesd204_block** (please refer to the jesd204_block created within IP Integrator for a more detailed view).



The JESD204 block contains:

- Xilinx JESD204B v6.1 TX and RX IP configured with 8-lanes per link and shared logic in example design. The JESD204B TX and RX cores are configured and monitored via corresponding AXI4-Lite management interfaces [2].
- An AXI GPIO interface providing block-level control & status.
- A JESD204B PHY v2.0 configured with both CPLL's and QPLL supported. This IP implements the Xilinx GTX transceiver logic and control interfaces. The JESD204B PHY core has an AXI register interface to allow the GTX transceiver and PLL parameters to be reprogrammed to dynamically change the supported line rate.

  The GTX clocking attributes corresponding to the supported frequency bands (see register CNFG_JESD_RATE) are listed in Appendix 2.

© Copyright 2016 Xilinx

Data Block

The **data_block** implements the data interface to the system. A FIFO is in included for each JESD204B RX lane and a RAM block for each JESD204B TX lane see figure 5 for a block diagram of the **data_block** internals.

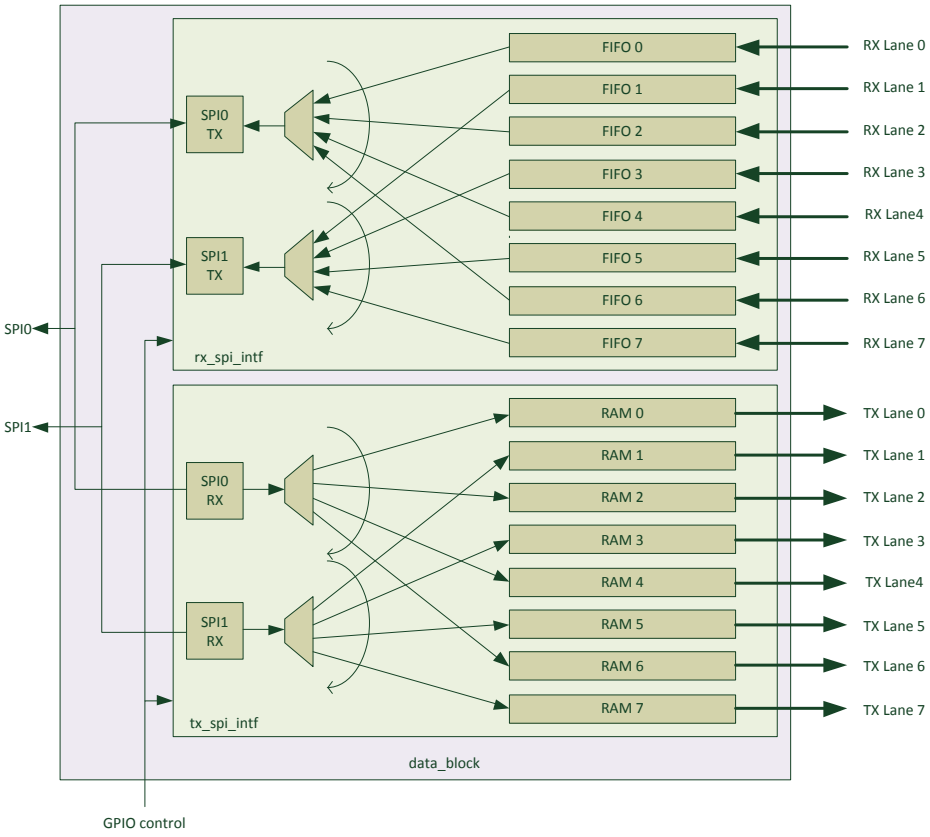


Figure 5 RX sample FIFO logic

Figure 6 shows how the JESD204B RX and TX lanes are connected to capture FIFO's and RAM's Figure 6 also shows how the FIF0's and RAMs are connected to the SPI ports. This split multiplexed arrangement allows the required sample data bandwidth to be shared between the two SPI ports. Each 32-bit SPI read or write causes the read or write multiplexor logic to advance to the next FIFO or RAM in the sequence. Individual enable bits are provided to control how many lanes are active and therefore the sequencing of the two read or write multiplexor's (See register RX_CNFG and TX_CNFG in appendix 1).

Capture and playback control of the data_block is controlled from HSDC Pro by writing to registers RX_CTRL and TX_CTRL whilst monitoring the Sample Block Status bits in the Status Register.

© Copyright 2016 Xilinx

Directory Structure

./hdl

Contains all the project hdl sources in individual directories.

./hdl/data_block/

Contains the source hdl for the data_block section of the design.

./hdl/jesd204_extras/

Contains the source hdl for the control & status GPIO interface and core clock divider logic.

./hdl/plat_id/

Contains the source hdl for the platform identification module used in the design.

./script

Contains all the scripts and sub-scripts used to generate IP, package sub-blocks and build the design hierarchy in IPI see the script section of this document for an overview.

./sw_src

Contains the source code for the Microblaze software running in this reference design. See the software section in this document for an overview.

./constraints

Contains the constraints files used to build this design for the supported boards.

After running the build_it.tcl script you will find the following directories have been created.

./local_IP_<BOARD>/projects/

This directory contains local projects generated to create packaged IP for use in the main project. The directory name contains the name of the board that was selected to build for (default VC707).

./local_IP_<BOARD>/repository/

This directory contains packaged IP for use in the main project. This packaged IP can be reused in other projects. The directory name contains the name of the board that was selected to build for (default VC707).

./proj_<BOARD>/

This directory contains the actual reference design project. The directory name contains the name of the board that was selected to build for (default VC707).

© Copyright 2016 Xilinx

Scripts

The scripts directory contains all the script used to build the design from the sources provided.

build_it.tcl:

This script is the main build script for the whole project. This script contains parameters that control the build configuration and does the following.

- Creates the main Vivado project.
- Creates directory ***./local_IP_<BOARD>/projects/*** to hold local projects created to package IP for use in the main project.
- Creates directory ***./local_IP_<BOARD>/repository/*** to hold locally created packaged IP.
- Calls script ***gen_and_pkg_data_block_IP.tcl*** (see description that follows).
- Calls script ***gen_and_pkg_single_file_module_IP.tcl*** to package rtl block ***jesd_extras***.
- Calls script ***gen_and_pkg_single_file_module_IP.tcl*** to package rtl block ***plat_id***.
- Adds the local packaged IP repository ***./local_IP_<BOARD>/repository/*** to the main project.
- Calls script ***create_block_diag.tcl*** which creates the IPI block diagram.
- Synthesizes the design.
- Implements the design and produces a bitstream.
- Exports the implemented design to the Xilinx SDK.

***gen_and_pkg_data_block_IP.tcl*:**

This sub-script creates and packages a project for the data_block hdl to use as a block in IP integrator as follows:

- Creates a Vivado project ***./local_IP/projects/data_block/data_block.xpr.***
- Imports hdl files from the ***./hdl/data_block/*** directory.
- Packages the project using IP packager.
- Copies the packaged project to the local repository directory ***./local_IP/repository/.***

© Copyright 2016 Xilinx

Software

The source code for the software running on the MicroBlaze processor is included for reference. The following gives a brief overview of the functionality.

tsw.c This is the main source file that initializes the system and then runs in a loop processing received commands.

The commands are received over the SPI control port. An SPI Interrupt service routine places the commands into a software queue for the main loop to process.

version.h This files contains the software version number.

global.h This file contains the register address definitions for the SPI control interface.

tsw_spi.c and ***tsw_spi.h*** These files are the source files for the SPI interface functions.

queue.c and ***queue.h*** These files are the source files for the command queue.

data_block.c and ***data_block.h*** These files are the source files for the control of the data_block.

jesd204.c and ***jesd204.h*** These files are the source files for the control of the jesd204 blocks.

clocks.c and ***clocks.h*** These files are the source files used for programming the line rate of the transceivers used by the JESD204B cores.

© Copyright 2016 Xilinx

Appendix 1 Register Map

This section details the map of the register interface that is accessed via the SPI control port SPI2 from the host PC. The register map is split into two sections. A single 32-bit read only status register and an addressable write only register space. The address range of the writeable register space is 30-bits. The 30-bit address space is further broken down into banks 1MByte in size. For simplicity all addressing is performed using bytes, but only 32-bit word writes are possible. The following simple protocol is used for communication.

- SPI configured for 32bit word transfers
- Two 32-bit word transfers are required per control write
- The first word constitutes a 2-bit command and a 30-bit address, in the register map, to be written to
- The second word constitutes the data payload to be written
- A single 32 bit status word is returned with each transfer. This status word communicates the basic state of the system back to the host PC.

Figure A1.1 shows pictorially how this SPI interface operates.

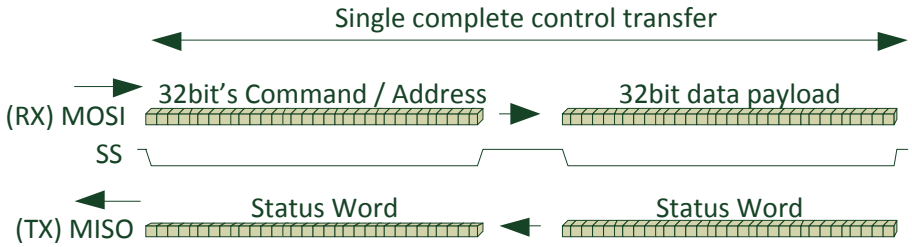


Figure A1.1 SPI control interface transfer

The 2-bit command has been included to accommodate possible future expansion of this interface beyond the basic write functionality define herein. The default command is "00" which corresponds to a write command.

© Copyright 2016 Xilinx

Read Only Status Register

This section details the bit definitions for the single read back word on the SPI2 port. This is a single 32bit register to provide feedback on the state of the system. This word is returned to the master during every 32-bit SPI write (It is therefore returned twice during a complete 64bit write access).

Offset: N/A	Name: STATUS

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OTHER STATUS							TX LINK STATUS				RX LINK STATUS				DATA BLOCK STATUS								IDENTITY REGISTER							
PHASE	Command Queue fill level							TX_RST_DONE	TX_RST_GT	TX_RST_CORE	TX_SYNC	RX_RST_DONE	RX_RST_GT	RX_RST_CORE	RX_SYNC			TX_EMPTY	TX_ERROR		RX_FULL	RX_EMPTY	RX_ERROR	PLATFORM ID				VERSION			

Bits	Purpose
31	PHASE This bit can be used by the SPI master controller to determine the current phase of a write access. This enables the master to confirm the communication with the slave is synchronized. 0 = Address phase current (Data phase next) 1 = Data phase current (Address phase next)
30:24	Command Queue Fill level. These bits indicate how many commands are in the queue waiting to be processed. To ensure the last command written has been processed. Poll address 0 until these bits return 0.
23	TX_RST_DONE This bit indicates that the reset TX sequence is complete 1 = Complete 0 = Not Complete
22	TX_RST_GT This bit indicates the state of the TX transceiver reset bit. 1 = In reset 0 = Not in reset
21	TX_RST_CORE This bit indicates the state of the TX JESD204B core reset bit. 1 = In reset 0 = Not in reset
20	TX_SYNC This bit indicates the state of the JESD204B SYNC input to the transmitter
19	RX_RST_DONE This bit indicates that the reset RX sequence is complete 1 = Complete 0 = Not Complete
18	RX_RST_GT This bit indicates the state of the RX transceiver reset bit. 1 = In reset 0 = Not in reset
17	RX_RST_CORE This bit indicates the state of the RX JESD204B core reset bit. 1 = In reset

© Copyright 2016 Xilinx

	0 = Not in reset
16	RX_SYNC This bit indicates the state of the JESD204B SYNC output from the Receiver
15-14	Unused
13	TX_EMPTY
12	TX_ERROR
11	Unused
10	RX_FULL
9	RX_EMPTY
8	RX_ERROR
7-4	PLATFORM ID: A 4 bit identifier unique per platform.
3:0	VERSION A 4 bit version number.

Writeable bank addressing

The following table shows 1Mbyte bank allocations

Bank Number	Base Address	Purpose
0	0x00000000	Sub-system control. This register bank is used for overall control the sub-system.
1	0x00100000	JESD204B receiver IP core. This register bank is used to write to the JESD204B receiver IP register interface.
2	0x00200000	JESD204B transmitter IP core. This register bank is used to write to the JESD204B transmitter IP register interface.

© Copyright 2016 Xilinx

Bank 0 Registers

This section details the registers used to configure and control the JESD204B IP sub-system. All addresses are specified as offsets to bank 0 base address.

	Control Registers
Address Offset	**Description**
0x0	Poll Status: A write to this address updates the Status Register and does not queue a command.
0x4	RESET
0x8	CTRL_GENERAL This register is for general control of the data sample block
0x10	CTRL_RX This register is for control of the RX path of the data sample block
0x14	CNFG_RX This register is for configuration of the RX path of the data sample block
0x20	CTRL_TX This register is for control of the TX path of the data sample block
0x24	CNFG_TX This register is for configuration of the TX path of the data sample block
0x40	CTRL_JESD Reset and reprogram PLL controls
0x44	CNFG_JESD_RATE Frequency band and clock multiplexor control register.
0x48	CNFG_JESD_REG1 Mode configuration register.
0x4C	CNFG_JESD_REG2 Transceiver parameter configuration register.

© Copyright 2016 Xilinx

Offset: 0x4	Name: RESET

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																															SOFT RESET

Bits	Purpose
31-1	Unused
0	SOFT RESET: Write 1 to this bit to force a soft reset of the MicroBlaze sub-system.

set: 0x8	Name: CTRL_GENERAL

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																															ADC_NOT_DAC

Bits	Purpose
31-1	Unused
0	ADC_NOT_DAC: This register bit controls the direction of the sample data block SPI ports (SPI ports 0 and 1). 0=Write to JESD204B TX RAM's 1=Read from JESD204B RX FIFO's

© Copyright 2016 Xilinx

Offset: 0x10	Name: CTRL_RX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																													STORE	FLUSH	

Bits	Purpose
31-3	Unused
2	STORE (1) 0 = Idle 1 = Start capturing link data into FIFO's. Data will be captured into the FIFO's for the lanes enables in register bits SPI0 _LANE_ENA and SPI1 _LANE_ENA
1	FLUSH (1) 0 = Normal operation 1 = Flush the RX pipeline. This bit must be held until status bit EX_EMPTY is asserted.
0	Unused

Notes:

1. You may only set one bit at any point in time and you must clear any set bits before setting another bit. (eg you cannot clear FLUSH and set STORE at the same time. You must write zero to this register to clear FLUSH before writing again to set STORE

© Copyright 2016 Xilinx

Offset: 0x14	Name: CNFG_RX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																								SPI1_LANE_ENA				SPI0_LANE_ENA			

Bits	Purpose
31-5	Unused
7-4	SPI1 _LANE_ENA (1) Individual enable bits for the SPI1 accessible lanes. These bits control which FIFO's get enabled for both capture and read out via SPI port 1. 0000 = No lanes enabled for SPI port 1 0001 = lane 1 enabled 0010 = lane 3 enabled 0100 = lane 5 enabled 1000 = lane 7 enabled
3-0	SPI0 _LANE_ENA (1) Individual enable bit for the SPI0 accessible lanes. These bits control which FIFO's get enabled for both capture and read out via SPI port 0. 0000 = No lanes enabled 0001 = lane 0 enabled 0010 = lane 2 enabled 0100 = lane 4 enabled 1000 = lane 6 enabled

Notes:

1. After configuring the lanes to be enabled for writing to over SPI. The FIFO interface must be cleared by performing a flush using register CTRL_RX.

© Copyright 2016 Xilinx

Offset: 0x20	Name: CTRL_TX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																												PLAY	STORE	FLUSH	

Bits	Purpose
31-4	Unused
3	PLAY (1) 0 = IDLE 1 = Enable output from data samples block into JESD204B TX core.
2	STORE (1) 0 = Idle and reset RAM write addresses. 1 = Start storing SPI sample data into RAM's. Data will be captured into the RAM's for the lanes enables in register bits SPI0 _LANE_ENA and SPI1 _LANE_ENA
1	FLUSH (1) 0 = Normal operation 1 = Flush the TX pipeline. This bit must be held until status bit TX_EMPTY is asserted.
0	Unused

Notes:

1. You may only set one bit at any point in time and you must clear any set bits before setting another bit. (eg you cannot clear FLUSH and set STORE at the same time. You must write zero to this register to clear FLUSH before writing again to set STORE

© Copyright 2016 Xilinx

Offset: 0x24	Name: CNFG_TX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																								SPI1_LANE_ENA				SPI0_LANE_ENA			

Bits	Purpose
31-8	Unused
7-4	SPI1 _LANE_ENA (2) Individual enable bit for SPI1 accessible lanes. These bits control which ram blocks get enabled for write via SPI port 1. 0000 = No lanes enabled for SPI port 1 0001 = lane 1 enabled 0010 = lane 3 enabled 0100 = lane 5 enabled 1000 = lane 7 enabled
3-0	SPI0 _LANE_ENA (2) Individual enable bit for SPI0 accessible lanes. These bits control which ram blocks get enabled for write via SPI port 0 0000 = No lanes enabled 0001 = lane 0 enabled 0010 = lane 2 enabled 0100 = lane 4 enabled 1000 = lane 6 enabled

Notes:

1. After selecting Pattern or Data output the PLAY bit must be set in register CTRL_TX to enable the output.
2. After configuring the lanes to be enabled for writing to over SPI. The RAM interface must be cleared by performing a flush using register CTRL_RX.

© Copyright 2016 Xilinx

Offset: 0x40	Name: CTRL_JESD

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																														DRPGO	JESD_RESET

Bits	Purpose
31-2	Unused
1	DRPGO: Write 1 to this bit to initiate reprogramming of the line rate settings (1)(2)(3).
0	JESD_RESET: Write 1 to this bit put the JESD204 cores and the transceivers into reset state.

Notes:

1. Registers CNFG_JESD_RATE, CNFG_JESD_REG1 and CNFG_JESD2 must be set before setting this bit.
2. JESD_RESET must be asserted before and while this bit is set.
3. Full setup process is as follows:
   - Assert JESD_RESET
   - Program CNFG_JESD_RATE, CNFG_JESD_REG1 and CNFG_JESD2.
   - Program RX core via BANK 1 or TX core registers via BANK 2
   - Assert DRPGO, with JESD_RESET still asserted.
   - Clear JESD_RESET.
   - Verify TX or RX _RST_DONE and _SYNC.

   Now the DATA_BLOCK may be used.

© Copyright 2016 Xilinx

Offset: 0x44	Name: CNFG_JESD_RATE

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BYPASS															DIV4_CLK									FULL_BAND				SUB BAND			

Bits	Purpose				
31	BYPASS: Must be set to zero.				
30-17	Unused				
16	DIV4_CLK: 0 = LMK04828 clk. 1 = Use internal LMK04828 clock divided by 4				
15:8	Unused				
7:0	FULL_BAND:	Range	Core Clk select(1)	Line rate / Ref Clk (2)	Ref Clk / Core Clk (3)
	0 =	1.0G -> 1.6G	DIV4_CLK	10	1
	1 =	1.6G -> 3.2G	DIV4_CLK	10	1
	2 =	3.2G -> 6.6G	LMK04828	20	2
	3 =	6.6G -> 8.0G	LMK04828	20	2
	4 =	9.8G -> 10.3G	LMK04828	20	2
3-0	SUB_BAND: Unused at this time.				

1. Set DIV4_CLK in this register.
2. At low rates Ref Clk = (Line Rate / 10). At high rates Ref Clk = (Line Rate / 20) to avoid violating max Ref Clock rate.
3. Set to 2 when the required core clock is within the LMK04828 range. At lower rates the Core Clk divider setting must equal the Ref Clock divider setting. The additional required core clock division is then done within the FPGA.

© Copyright 2016 Xilinx

Offset: 0x48	Name: CNFG_JESD_REG1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
															INVSYNC															MODE	

Bits	Purpose
31-17	Unused
16	INVSYNC: Write 1 to invert the JESD204B SYNC output. This is required for ADS42JB69.
15:2	Unused
1:0	MODE: 0 = OFF 1 = ADC 2 = DAC 3 = ADC and DAC

Offset: 0x4C	Name: CNFG_JESD_REG2

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		GT_TXPOSTCURSER					GT_TXPRECURSER					GT_TXDIFFCTRL				GT_TXPOLARITY								GT_RXPOLARITY							

Bits	Purpose
31-30	Unused
29-25	GT_TXPOSTCURSER See [3] for explanation.
24-20	GT_TXPOSTCURSER See [3] for explanation.
19-16	GT_TXPRECURSER See [3] for explanation.
15-8	GT_TXDIFFCTRL See [3] for explanation. Note 1-bit per transceiver lane.
7-0	GT_RXPOLARITY See [3] for explanation. Note 1-bit per transceiver lane.

© Copyright 2016 Xilinx

Bank 1 Registers

This section details the registers used to configure and control the JESD204B Receiver IP. All addresses are specified as offsets from bank 1 base address.

See [2] register addresses and definitions.

Bank 2 Registers

This section details the registers used to configure and control the JESD204B TX IP. All addresses are specified as offsets from bank 2 base address.

See [2] for register addresses and definitions.

© Copyright 2016 Xilinx

Appendix 2 GTX DRP Attributes

The following table lists the GTX2_COMMON and GTX2_CHANNEL attributes programmed by the configuration software when the corresponding FULL_BAND frequency bands are selected (see register CNFG_JESD_RATE). The table shows the attribute encoding of the selected DRP parameters; refer to [3] for the DRP encoded values used by the configuration software.

Note that only a subset of the DRP accessible registers is shown for both GTX configuration ports. This subset contains attributes that require dynamic updates to implement the line rate switching capability or are directly linked to the selected clocking scheme. In order to meet the combination of frequency constraints for the GTX reference clocks, JESD204B IP core clock, and the EVM's LMK04828 clock jitter cleaner, the expected reference clock rate is set to 2x FPGA JESD204B IP clock frequency(1) at line rates above 3.2Gbps and 1x for the lower supported bands.

Line Rate(2)	1.0G -> 1.6G	1.6G -> 3.2G	3.2G -> 6.6G	6.6G -> 8.0G	9.8G -> 10.3G
GTX2_COMMON DRP					
QPLL_FBDIV	20	20	20	20	20
QPLL_REFCLK_DIV	1	1	1	1	1
QPLL_CFG	0x06801C1	0x06801C1	0x06801C1	0x06801C1	0x0680181
GTX2_CHANNEL DRP					
CPLL_FBDIV	4	2	2	4	4
CPLL_FBDIV_45	5	5	5	5	5
CPLL_REFCLK_DIV	1	1	1	1	1
RXOUT_DIV	4	2	1	1	1
TXOUT_DIV	4	2	1	1	1
RX_CLK25_DIV	6	12	10	15	20
TX_CLK25_DIV	6	12	10	15	20
RXCDR_CFG	0x03_0000_23FF_4008_0020	0x03_0000_23FF_4020_0020	0x03_0000_23FF_2040_0020	0x0B_0000_23FF_1040_0020	0x0B_0000_23FF_1040_0020

1. Xilinx JESD204B IP tx_core_clock & rx_core_clk frequency = serial line rate / 40. Refer to [2] for additional information.
2. Xilinx JESD204B IP supports line rates from 1Gbps; GTXE2 transceivers are rated up to 10.3125Gbps for the targeted Kintex-7 (speed grade -2) devices with a frequency band gap at 8-9.8Gbps [4].

© Copyright 2016 Xilinx