ad9361x2_pl_wrapper.v

AUTHORS

JAY CONVERTINO

DATES

2023/11/02

INFORMATION

Brief

AD9361x2 core and support core wrapper.

License MIT

Copyright 2023 Jay Convertino

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

ad9361x2_pl_wrapper

```
module ad9361x2_pl_wrapper #(
parameter
FPGA_TECHNOLOGY
 =
0,
parameter
FPGA_FAMILY
 =
0,
parameter
SPEED_GRADE
```

```
 =
0,
parameter
DEV_PACKAGE
 =
0,
parameter
ADC_INIT_DELAY
 =
23,
parameter
DAC_INIT_DELAY
 =
0,
parameter
DELAY_REFCLK_FREQUENCY
 =
200,
parameter
DMA_AXI_PROTOCOL_TO_PS
 =
1,
parameter
AXI_DMAC_ADC_ADDR
 =
32'h7C400000,
parameter
AXI_DMAC_DAC_ADDR
 =
32'h7C420000,
parameter
AXI_AD9361_0_ADDR
 =
32'h79020000,
parameter
AXI_AD9361_1_ADDR
 =
32'h79040000
) ( input axi_aclk, input axi_aresetn, input s_axi_awvalid, input [31:0] s_a
```

AD9361x2 core and support core wrapper.

Parameters

FPGA_TECHNOLOGY parameter	Type of FPGA, such as Ultrascale, Arria 10. 1 is for 7 series.
FPGA_FAMILY parameter	Sub type of fpga, such as GX, SX, etc. 4 is for zynq.
SPEED_GRADE parameter	Number that corresponds to the ships recommeneded speed. 20 is for -2.
DEV_PACKAGE parameter	Specify a number that is equal to the manufactures package. 3 is for ff.
DELAY_REFCLK_FREQUENCY parameter	Reference clock frequency used for ad_data_in instances
ADC_INIT_DELAY parameter	Initial Delay for the ADC
DAC_INIT_DELAY parameter	Initial Delay for the DAC
DMA_AXI_PROTOCOL_TO_PS parameter	Select DMA AXI standard, 1 = AXI3, 0 = AXI4

AXI_DMAC_ADC_ADDR parameter	Set ADC AXI lite address.
AXI_DMAC_DAC_ADDR parameter	Set DAC AXI lite address.
AXI_AD9361_0_ADDR parameter	Set AD9361 0 AXI lite address.
AXI_AD9361_1_ADDR parameter	Set AD9361 1 AXI lite address.

Ports

axi_aclk	AXI Lite control bus
axi_aresetn	AXI Lite control bus
s_axi_awvalid	AXI Lite control bus
s_axi_awaddr	AXI Lite control bus
s_axi_awready	AXI Lite control bus
s_axi_awprot	AXI Lite control bus
s_axi_wvalid	AXI Lite control bus
s_axi_wdata	AXI Lite control bus
s_axi_wstrb	AXI Lite control bus
s_axi_wready	AXI Lite control bus
s_axi_bvalid	AXI Lite control bus
s_axi_bresp	AXI Lite control bus
s_axi_bready	AXI Lite control bus
s_axi_arvalid	AXI Lite control bus
s_axi_araddr	AXI Lite control bus
s_axi_arready	AXI Lite control bus
s_axi_arprot	AXI Lite control bus
s_axi_rvalid	AXI Lite control bus
s_axi_rready	AXI Lite control bus
s_axi_rresp	AXI Lite control bus
s_axi_rdata	AXI Lite control bus
adc_dma_irq	fmcomms5 ADC irq
dac_dma_irq	fmcomms5 DAC irq
delay_clk	fmcomms5 delay clock
rx_clk_in_0_p	fmcomms5 0 rx clk
rx_clk_in_0_n	fmcomms5 0 rx clk
rx_frame_in_0_p	fmcomms5 0 rx frame
rx_frame_in_0_n	fmcomms5 0 rx frame
rx_data_in_0_p	fmcomms5 0 rx data
rx_data_in_0_n	fmcomms5 0 rx data
tx_clk_out_0_p	fmcomms5 0 tx clk
tx_clk_out_0_n	fmcomms5 0 tx clk
tx_frame_out_0_p	fmcomms5 0 tx frame
tx_frame_out_0_n	fmcomms5 0 tx frame

tx_data_out_0_p	fmcomms5 0 tx data
tx_data_out_0_n	fmcomms5 0 tx data
txnrx_0	fmcomms5 0 txnrx
enable_0	fmcomms5 0 enable
up_enable_0	fmcomms5 0 enable input
up_txnrx_0	fmcomms5 0 txnrx select input
tdd_sync_0_t	fmcomms5 0 TDD sync i/o
tdd_sync_0_i	fmcomms5 0 TDD sync i/o
tdd_sync_0_o	fmcomms5 0 TDD sync i/o
rx_clk_in_1_p	fmcomms5 1 rx clk
rx_clk_in_1_n	fmcomms5 1 rx clk
rx_frame_in_1_p	fmcomms5 1 rx frame
rx_frame_in_1_n	fmcomms5 1 rx frame
rx_data_in_1_p	fmcomms5 1 rx data
rx_data_in_1_n	fmcomms5 1 rx data
tx_clk_out_1_p	fmcomms5 1 tx clk
tx_clk_out_1_n	fmcomms5 1 tx clk
tx_frame_out_1_p	fmcomms5 1 tx frame
tx_frame_out_1_n	fmcomms5 1 tx frame
tx_data_out_1_p	fmcomms5 1 tx data
tx_data_out_1_n	fmcomms5 1 tx data
txnrx_1	fmcomms5 1 txnrx
enable_1	fmcomms5 1 enable
up_enable_1	fmcomms5 1 enable input
up_txnrx_1	fmcomms5 1 txnrx select input
tdd_sync_1_t	fmcomms5 1 TDD sync i/o
tdd_sync_1_i	fmcomms5 1 TDD sync i/o
tdd_sync_1_o	fmcomms5 1 TDD sync i/o
m_axi_aclk	DMA Clock
m_axi_aresetn	DMA Negative Reset
adc_m_dest_axi_awaddr	fmcomms5 ADC DMA
adc_m_dest_axi_awlen	fmcomms5 ADC DMA
adc_m_dest_axi_awsize	fmcomms5 ADC DMA
adc_m_dest_axi_awburst	fmcomms5 ADC DMA
adc_m_dest_axi_awprot	fmcomms5 ADC DMA
adc_m_dest_axi_awcache	fmcomms5 ADC DMA
adc_m_dest_axi_awvalid	fmcomms5 ADC DMA
adc_m_dest_axi_awready	fmcomms5 ADC DMA
adc_m_dest_axi_wdata	fmcomms5 ADC DMA
adc_m_dest_axi_wstrb	fmcomms5 ADC DMA
adc_m_dest_axi_wready	fmcomms5 ADC DMA
adc_m_dest_axi_wvalid	fmcomms5 ADC DMA

adc_m_dest_axi_wlast	fmcomms5 ADC DMA
adc_m_dest_axi_bvalid	fmcomms5 ADC DMA
adc_m_dest_axi_bresp	fmcomms5 ADC DMA
adc_m_dest_axi_bready	fmcomms5 ADC DMA
dac_m_src_axi_arready	fmcomms5 DAC DMA
dac_m_src_axi_arvalid	fmcomms5 DAC DMA
dac_m_src_axi_araddr	fmcomms5 DAC DMA
dac_m_src_axi_arlen	fmcomms5 DAC DMA
dac_m_src_axi_arsize	fmcomms5 DAC DMA
dac_m_src_axi_arburst	fmcomms5 DAC DMA
dac_m_src_axi_arprot	fmcomms5 DAC DMA
dac_m_src_axi_arcache	fmcomms5 DAC DMA
dac_m_src_axi_rdata	fmcomms5 DAC DMA
dac_m_src_axi_rready	fmcomms5 DAC DMA
dac_m_src_axi_rvalid	fmcomms5 DAC DMA
dac_m_src_axi_rresp	fmcomms5 DAC DMA
dac_m_src_axi_rlast	fmcomms5 DAC DMA

INSTANTIANTED MODULES

inst_axi_ad9361_0

```
axi_ad9361 #(
ID(0),
MODE_1R1T(0),
FPGA_TECHNOLOGY(FPGA_TECHNOLOGY),
FPGA_FAMILY(FPGA_FAMILY),
SPEED_GRADE(SPEED_GRADE),
DEV_PACKAGE(DEV_PACKAGE),
TDD_DISABLE(0),
PPS_RECEIVER_ENABLE(0),
CMOS_OR_LVDS_N(0),
ADC_INIT_DELAY(ADC_INIT_DELAY),
ADC_DATAPATH_DISABLE(0),
ADC_USERPORTS_DISABLE(0),
ADC_DATAFORMAT_DISABLE(0),
ADC_DCFILTER_DISABLE(0),
ADC_IQCORRECTION_DISABLE(0),
DAC_INIT_DELAY(DAC_INIT_DELAY),
DAC_CLK_EDGE_SEL(0),
DAC_IODELAY_ENABLE(0),
DAC_DATAPATH_DISABLE(0),
DAC_DDS_DISABLE(0),
DAC_DDS_TYPE(1),
DAC_DDS_CORDIC_DW(14),
DAC_DDS_CORDIC_PHASE_DW(13),
DAC_USERPORTS_DISABLE(0),
DAC_IQCORRECTION_DISABLE(0),
IO_DELAY_GROUP("dev_0_if_delay_group"),
MIMO_ENABLE(0),
USE_SSI_CLK(1),
DELAY_REFCLK_FREQUENCY(DELAY_REFCLK_FREQUENCY),
RX_NODPA(0)
) inst_axi_ad9361_0 ( .rx_clk_in_p(rx_clk_in_0_p), .rx_clk_in_n(rx_clk_in_0_
```

Analog Devices ad9361 0 interface core

inst_axi_ad9361_1

```
axi_ad9361 #(
ID(1),
MODE_1R1T(0),
FPGA_TECHNOLOGY(FPGA_TECHNOLOGY),
FPGA_FAMILY(FPGA_FAMILY),
SPEED_GRADE(SPEED_GRADE),
DEV_PACKAGE(DEV_PACKAGE),
TDD_DISABLE(0),
PPS_RECEIVER_ENABLE(0),
CMOS_OR_LVDS_N(0),
ADC_INIT_DELAY(ADC_INIT_DELAY),
ADC_DATAPATH_DISABLE(0),
ADC_USERPORTS_DISABLE(0),
ADC_DATAFORMAT_DISABLE(0),
ADC_DCFILTER_DISABLE(0),
ADC_IQCORRECTION_DISABLE(0),
DAC_INIT_DELAY(DAC_INIT_DELAY),
DAC_CLK_EDGE_SEL(0),
DAC_IODELAY_ENABLE(0),
DAC_DATAPATH_DISABLE(0),
DAC_DDS_DISABLE(0),
DAC_DDS_TYPE(1),
DAC_DDS_CORDIC_DW(14),
DAC_DDS_CORDIC_PHASE_DW(13),
DAC_USERPORTS_DISABLE(0),
DAC_IQCORRECTION_DISABLE(0),
IO_DELAY_GROUP("dev_1_if_delay_group"),
MIMO_ENABLE(0),
USE_SSI_CLK(0),
DELAY_REFCLK_FREQUENCY(DELAY_REFCLK_FREQUENCY),
RX_NODPA(0)
) inst_axi_ad9361_1 ( .rx_clk_in_p(rx_clk_in_1_p), .rx_clk_in_n(rx_clk_in_1
```

Analog Devices ad9361 1 interface core

inst_adc_axi_dmac

```
axi_dmac #(
ID(0),
DMA_DATA_WIDTH_SRC(128),
DMA_DATA_WIDTH_DEST(64),
DMA_LENGTH_WIDTH(24),
DMA_2D_TRANSFER(0),
ASYNC_CLK_REQ_SRC(1),
ASYNC_CLK_SRC_DEST(1),
ASYNC_CLK_DEST_REQ(1),
AXI_SLICE_DEST(0),
AXI_SLICE_SRC(1),
                                                                          .
 SYNC_TRANSFER_START(1),
                                                                          .
 CYCLIC(0),
                                                                          .
 DMA_AXI_PROTOCOL_DEST(DMA_AXI_PROTOCOL_TO_PS),
                                                                          .
 DMA_AXI_PROTOCOL_SRC(1),
                                                                          .
 DMA_TYPE_DEST(0),
                                                                          .
 DMA_TYPE_SRC(1),
                                                                          .
 DMA_AXI_ADDR_WIDTH(32),
                                                                          .
 MAX_BYTES_PER_BURST(128),
                                                                          .
 FIFO_SIZE(8),
                                                                          .
 AXI_ID_WIDTH_SRC(6),
                                                                          .
 AXI_ID_WIDTH_DEST(6),
                                                                          .
 DMA_AXIS_ID_W(8),
                                                                          .
 DMA_AXIS_DEST_W(4),
                                                                          .
 DISABLE_DEBUG_REGISTERS(0),
                                                                          .
 ENABLE_DIAGNOSTICS_IF(0),
                                                                          .
 ALLOW_ASYM_MEM(1),
                                                                          .
 CACHE_COHERENT_DEST(1)
) inst_adc_axi_dmac ( .s_axi_aclk(axi_aclk), .s_axi_aresetn(axi_aresetn), .
```

Analog Devices DMA for AD9361 ADC

inst_dac_axi_dmac

```
 axi_dmac #(
                                                                          .
 ID(0),
                                                                          .
 DMA_DATA_WIDTH_SRC(64),
                                                                          .
 DMA_DATA_WIDTH_DEST(128),
                                                                          .
 DMA_LENGTH_WIDTH(24),
                                                                          .
 DMA_2D_TRANSFER(0),
                                                                          .
 ASYNC_CLK_REQ_SRC(1),
                                                                          .
 ASYNC_CLK_SRC_DEST(1),
                                                                          .
 ASYNC_CLK_DEST_REQ(1),
                                                                          .
 AXI_SLICE_DEST(1),
                                                                          .
 AXI_SLICE_SRC(0),
                                                                          .
 SYNC_TRANSFER_START(0),
                                                                          .
 CYCLIC(1),
                                                                          .
 DMA_AXI_PROTOCOL_DEST(1),
                                                                          .
 DMA_AXI_PROTOCOL_SRC(DMA_AXI_PROTOCOL_TO_PS),
                                                                          .
 DMA_TYPE_DEST(1),
                                                                          .
 DMA_TYPE_SRC(0),
                                                                          .
 DMA_AXI_ADDR_WIDTH(32),
                                                                          .
 MAX_BYTES_PER_BURST(128),
                                                                          .
 FIFO_SIZE(8),
                                                                          .
 AXI_ID_WIDTH_SRC(6),
                                                                          .
 AXI_ID_WIDTH_DEST(6),
                                                                          .
 DMA_AXIS_ID_W(8),
                                                                          .
 DMA_AXIS_DEST_W(4),
                                                                          .
 DISABLE_DEBUG_REGISTERS(0),
                                                                          .
 ENABLE_DIAGNOSTICS_IF(0),
                                                                          .
 ALLOW_ASYM_MEM(1),
                                                                          .
 CACHE_COHERENT_DEST(0)
) inst_dac_axi_dmac ( .s_axi_aclk(axi_aclk), .s_axi_aresetn(axi_aresetn), .s
```

Analog Devices DMA for AD9361 DAC

inst_adc_cpack

```
 util_cpack2_axis #(
                                                                          .
 NUM_OF_CHANNELS(8),
                                                                          .
 SAMPLES_PER_CHANNEL(1),
                                                                          .
 SAMPLE_DATA_WIDTH(16)
) inst_adc_cpack ( .clk(d_clk), .reset(p_reset), .enable_0(fifo_adc_enable_
```

Analog Devices Utility to take ad9361 data and pack it to a AXIS bus for the ADC

inst_dac_cpack

Analog Devices Utility to take ad9361 data and unpack from the AXIS bus to the DAC

inst_dac_fifo

```
 util_rfifo #(
                                                                          .
 NUM_OF_CHANNELS(8),
                                                                          .
 DIN_DATA_WIDTH(16),
                                                                          .
 DOUT_DATA_WIDTH(16),
                                                                          .
 DIN_ADDRESS_WIDTH(4)
 ) inst_dac_fifo ( .din_rstn(p_aresetn), .din_clk(d_clk), .din_enable_0(fif
```

Analog Devices FIFO for AD9361 DAC BUS

inst_adc_fifo

```
util_wfifo #(
                                                                         .
NUM_OF_CHANNELS(8),
                                                                         .
DIN_DATA_WIDTH(16),
                                                                         .
DOUT_DATA_WIDTH(16),
                                                                         .
DIN_ADDRESS_WIDTH(4)
) inst_adc_fifo ( .din_rst(ad_reset_o), .din_clk(l_clk), .din_enable_0(adc_
```

Analog Devices FIFO for AD9361 ADC BUS

inst_clkdiv

```
util_clkdiv #(
                                                                         .
SIM_DEVICE(SIM_DEVICE)
) inst_clkdiv ( .clk(l_clk), .clk_sel(adc_r1_mode_0 & dac_r1_mode_0 & adc_
```

Analog Devices Clock Divider with select

isnt_util_tdd_sync_0

```
util_tdd_sync #(
                                                                         .
TDD_SYNC_PERIOD(100000000)
) isnt_util_tdd_sync_0 ( .clk(axi_aclk), .rstn(axi_aresetn), .sync_mode(td
```

Analog Devices tdd sync utility

isnt_util_tdd_sync_1

```
util_tdd_sync #(
                                                                         .
TDD_SYNC_PERIOD(100000000)
) isnt_util_tdd_sync_1 ( .clk(axi_aclk), .rstn(axi_aresetn), .sync_mode(td
```

Analog Devices tdd sync utility

inst_ad_reset

```
ad_rst inst_ad_reset (
                                                                        .
rst_async(~axi_aresetn),
                                                                        .
clk(d_clk),
                                                                        .
rstn(p_aresetn),
                                                                        .
rst(p_reset)
)
```

Analog Devices reset sync

inst_axilxbar

```
axilxbar #(
                                                                        .
C_AXI_DATA_WIDTH(32),
                                                                        .
C_AXI_ADDR_WIDTH(32),
                                                                        .
NM(1),
                                                                        .
NS(4),
                                                                        .
SLAVE_ADDR({{AXI_DMAC_ADC_ADDR},{AXI_DMAC_DAC_ADDR},{AXI_AD9361_1_ADDR}, {
                                                                        .
SLAVE_MASK({{32'hFFFFF000},{32'hFFFFF000},{32'hFFFF0000}, {32'hFFFF0000}})
) inst_axilxbar ( .S_AXI_ACLK(axi_aclk), .S_AXI_ARESETN(axi_aresetn), .S_AX
```

AXI Lite crossbar for ADC DMA, DAC DMA, and AD9361 1/0 control registers.