FMCOMMS5

November 22, 2024

Jay Convertino

Contents

1 Usage

1.1 Introduction

The fmcomms5 project builds a FPGA base system for the fmcomms5 Analog Devices development boards. Project targets are listed in 3.3. The base IP for the Analog Devices parts are from the Analog Devices HDL repo. They have been converted into fusesoc cores and some modifications have been made. Modifications include making the AD-C/DAC routes both use AXIS out of the DMAs. The Intel FPGA targets now uses ad_data/ad_clock cores, clock select, and DC filter to reach functionally on par with Xilinx targets.

1.2 Dependencies

The following are the dependencies of the cores.

- fusesoc 2.X
- iverilog (simulation)
- cocotb (simulation)

1.2.1 fusesoc_info Depenecies

- zc706
  - AFRL:utility:xilinx_zc706_board_base:1.0.0
  - AFRL:utility:vivado_board_support_packages
  - AD:common:ad_iobuf:1.0.0
- zc706_bootgen
  - AFRL:utility:xilinx_zc706_boot_gen:1.0.0
- zc702
  - AFRL:utility:xilinx_zc702_board_base:1.0.0
  - AFRL:utility:vivado_board_support_packages
  - AD:common:ad_iobuf:1.0.0
- zc702_bootgen
  - AFRL:utility:xilinx_zc702_boot_gen:1.0.0
- zcu102
  - AFRL:utility:xilinx_zcu102_board_base:1.0.0

    - AFRL:utility:vivado_board_support_packages
- zcu102_bootgen
    - AFRL:utility:xilinx_zcu102_boot_gen:1.0.0
- dep
    - AD:RF_Transceiver:axi_ad9361:1.0.0
    - AD:utility:tdd_sync:1.0.0
    - AD:memory_controller:axi_dmac:1.0.0
    - AD:data_flow:util_cpack_axis:1.0.0
    - AD:data_flow:util_upack:2.0.0
    - AD:buffer:util_rfifo:1.0.0
    - AD:buffer:util_wfifo:1.0.0
    - AD:common:util_clkdiv:1.0.0
    - AD:common:ad_rst:1.0.0
    - AFRL:utility:tcl_helper_check:1.0.0
    - zipcpu:axi_lite:crossbar:1.0.0

2 Architecture

The project contains four wrappers

- **system_wrapper** Contains the top level project module and contains system_pl_wrapper and system_ps_wrapper.
- **system_pl_wrapper** Contains the AD9361 wrapper and any support IP's in the program logic.
- **ad9361_pl_wrapper** Contains all program logic IP's dealing with the AD9361x2.
- **system_ps_wrapper** Contains the processor system IP wrappers.

Please see 5 for more information per target.

3 Building

The all fmcomms5 core is written in Verilog 2001. They should synthesize in any modern FPGA software. The core comes as a fusesoc packaged core and can be included in any other core. Be sure to make sure you have meet the dependencies listed in the previous section.

3.1 fusesoc

Fusesoc is a system for building FPGA software without relying on the internal project management of the tool. Avoiding vendor lock in to Vivado or Quartus. These cores, when included in a project, can be easily integrated and targets created based upon the end developer needs. The core by itself is not a part of a system and should be integrated into a fusesoc based system. Simulations are setup to use fusesoc and are a part of its targets.

3.2 Source Files

3.2.1 fusesoc_info File List

- src_ad9361x2_pl
  - 'common/ad9361x2_pl_wrapper.v': 'file_type': 'verilogSource'
- zc706
  - 'zc706/system_constr.xdc': 'file_type': 'xdc'
  - 'zc706/system_wrapper.v': 'file_type': 'verilogSource'
  - 'zc706/system_pl_wrapper.v': 'file_type': 'verilogSource'
  - 'zc706/system_pl_gen.tcl': 'file_type': 'tclSource'
  - 'zc706/system_gen.tcl': 'file_type': 'tclSource'
- zc702
  - 'zc702/system_constr.xdc': 'file_type': 'xdc'
  - 'zc702/system_wrapper.v': 'file_type': 'verilogSource'
  - 'zc702/system_pl_wrapper.v': 'file_type': 'verilogSource'
  - 'zc702/system_pl_gen.tcl': 'file_type': 'tclSource'
  - 'zc702/system_gen.tcl': 'file_type': 'tclSource'
- zcu102
  - 'zcu102/system_constr.xdc': 'file_type': 'xdc'
  - 'zcu102/system_wrapper.v': 'file_type': 'verilogSource'
  - 'zcu102/system_pl_wrapper.v': 'file_type': 'verilogSource'
  - 'zcu102/system_pl_gen.tcl': 'file_type': 'tclSource'
  - 'zcu102/system_gen.tcl': 'file_type': 'tclSource'

3.3 Targets

3.3.1 fusesoc_info Targets

- default

  Info: Default target, do not use.

- zc706

  Info: zc706 target.

- zc706_bootgen

  Info: zc706 build with boot.bin output in BOOTFS folder.

- zc702

  Info: zc702 target.

- zc702_bootgen

  Info: zc702 build with boot.bin output in BOOTFS folder.

- zcu102

  Info: zcu102 target.

- zcu102_bootgen

  Info: zcu102 build with boot.bin output in BOOTFS folder.

3.4 Directory Guide

Below highlights important folders from the root of the directory.

1. **docs** Contains all documentation related to this project.
   - **manual** Contains user manual and github page that are generated from the latex sources.
2. **common** Contains source file wrapper for ad9361 core
3. **zc702** Contains source files for Xilinx zc702
4. **zc706** Contains source files for Xilinx zc706
5. **zcu102** Contains source files for Xilinx zcu102

4 Simulation

There is no simulation at the moment. This is dues to the AD9361 and ARM subsystems. Maybe a future addition with Vexriscv?

5 Module Documentation

There project has multiple modules. The targets are the top system wrappers.

- **ad9361 system pl**
- **zc702 system pl**
- **zc702 system**
- **zc706 system pl**
- **zc706 system**
- **zcu102 system pl**
- **zcu102 system**

The next sections document the module in great detail.

ad9361x2_pl_wrapper.v

AUTHORS

JAY CONVERTINO

DATES

2023/11/02

INFORMATION

Brief

AD9361x2 core and support core wrapper.

License MIT

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), .s
```

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_i
```

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(fifo
```

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_r
```

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(tdd
```

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(tdd
```

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}, {A
                                                                        .
SLAVE_MASK({{32'hFFFFF000},{32'hFFFFF000},{32'hFFFF0000}, {32'hFFFF0000}})
) inst_axilxbar ( .S_AXI_ACLK(axi_aclk), .S_AXI_ARESETN(axi_aresetn), .S_AXI
```

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

system_pl_wrapper.v

AUTHORS

JAY CONVERTINO

DATES

2023/11/02

INFORMATION

Brief

System wrapper for pl only for zc702 board.

License MIT

system_pl_wrapper

```
module system_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
) ( input axi_aclk, input axi_aresetn, input s_axi_awvalid, input [31:0] s_
```

System wrapper for pl only for zc702 board.

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. 10 is for -1.
DEV_PACKAGE parameter	Specify a number that is equal to the manufactures package. 14 is for cl.
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

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
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
iic_sda_fmc	i2c for fmc
iic_scl_fmc	i2c for fmc
iic2intc_irpt	i2c for fmc

INSTANTIANTED MODULES

iic_sda_iobuf

```
ad_iobuf #(
                                                                                .
DATA_WIDTH(1)
) iic_sda_iobuf ( .dio_t (sda_t), .dio_i (sda_o), .dio_o (sda_i), .dio_p (
```

Tristate i2c sda

iic_scl_iobuf

```
ad_iobuf #(
                                                                                .
DATA_WIDTH(1)
) iic_scl_iobuf ( .dio_t (scl_t), .dio_i (scl_o), .dio_o (scl_i), .dio_p (
```

Tristate i2c scl

inst_dma_rstgen

```
dma_rstgen inst_dma_rstgen (
                                                                                .
slowest_sync_clk(delay_clk),
                                                                                .
ext_reset_in(axi_aresetn),
                                                                                .
aux_reset_in(1'b1),
                                                                                .
mb_debug_sys_rst(1'b0),
                                                                                .
dcm_locked(1'b1),
                                                                                .
mb_reset(),
                                                                                .
bus_struct_reset(),
                                                                                .
peripheral_reset(),
                                                                                .
interconnect_aresetn(),
                                                                                .
peripheral_aresetn(m_axi_aresetn)
)
```

Generate a new DMA reset based on delay clock.

inst_ad9361x2_pl_wrapper

```
ad9361x2_pl_wrapper #(
                                                                                .
FPGA_TECHNOLOGY(FPGA_TECHNOLOGY),
                                                                                .
FPGA_FAMILY(FPGA_FAMILY),
```

```
                                                                                .
SPEED_GRADE(SPEED_GRADE),                                                       .
DEV_PACKAGE(DEV_PACKAGE),                                                       .
ADC_INIT_DELAY(ADC_INIT_DELAY),                                                 .
DAC_INIT_DELAY(DAC_INIT_DELAY),                                                 .
DELAY_REFCLK_FREQUENCY(DELAY_REFCLK_FREQUENCY)
) inst_ad9361x2_pl_wrapper ( .axi_aclk(axi_aclk), .axi_aresetn(axi_aresetn)
```

Module instance of inst_ad9361x2_pl_wrapper for the fmcomms5 device.

inst_axi_crossbar_pl

```
axi_crossbar_pl inst_axi_crossbar_pl (
                                                                                .
aclk(axi_aclk),                                                                 .
aresetn(axi_aresetn),                                                           .
s_axi_awaddr(s_axi_awaddr),                                                     .
s_axi_awprot(s_axi_awprot),                                                     .
s_axi_awvalid(s_axi_awvalid),                                                   .
s_axi_awready(s_axi_awready),                                                   .
s_axi_wdata(s_axi_wdata),                                                       .
s_axi_wstrb(s_axi_wstrb),                                                       .
s_axi_wvalid(s_axi_wvalid),                                                     .
s_axi_wready(s_axi_wready),                                                     .
s_axi_bresp(s_axi_bresp),                                                       .
s_axi_bvalid(s_axi_bvalid),                                                     .
s_axi_bready(s_axi_bready),                                                     .
s_axi_araddr(s_axi_araddr),                                                     .
s_axi_arprot(s_axi_arprot),                                                     .
s_axi_arvalid(s_axi_arvalid),                                                   .
s_axi_arready(s_axi_arready),                                                   .
s_axi_rdata(s_axi_rdata),                                                       .
s_axi_rresp(s_axi_rresp),                                                       .
s_axi_rvalid(s_axi_rvalid),                                                     .
s_axi_rready(s_axi_rready),                                                     .
m_axi_awaddr({iic_fmc_axi_awaddr, connect_axi_awaddr}),                         .
m_axi_awprot({iic_fmc_axi_awprot, connect_axi_awprot}),
```

```
 m_axi_awvalid({iic_fmc_axi_awvalid, connect_axi_awvalid}),
 m_axi_awready({iic_fmc_axi_awready, connect_axi_awready}),
 m_axi_wdata({iic_fmc_axi_wdata, connect_axi_wdata}),
 m_axi_wstrb({iic_fmc_axi_wstrb, connect_axi_wstrb}),
 m_axi_wvalid({iic_fmc_axi_wvalid, connect_axi_wvalid}),
 m_axi_wready({iic_fmc_axi_wready, connect_axi_wready}),
 m_axi_bresp({iic_fmc_axi_bresp, connect_axi_bresp}),
 m_axi_bvalid({iic_fmc_axi_bvalid, connect_axi_bvalid}),
 m_axi_bready({iic_fmc_axi_bready, connect_axi_bready}),
 m_axi_araddr({iic_fmc_axi_araddr, connect_axi_araddr}),
 m_axi_arprot({iic_fmc_axi_arprot, connect_axi_arprot}),
 m_axi_arvalid({iic_fmc_axi_arvalid, connect_axi_arvalid}),
 m_axi_arready({iic_fmc_axi_arready, connect_axi_arready}),
 m_axi_rdata({iic_fmc_axi_rdata, connect_axi_rdata}),
 m_axi_rresp({iic_fmc_axi_rresp, connect_axi_rresp}),
 m_axi_rvalid({iic_fmc_axi_rvalid, connect_axi_rvalid}),
 m_axi_rready({iic_fmc_axi_rready, connect_axi_rready})
)
```

Module instance of axi_crossbar_pl for the fmcomms5 device.

inst_axi_iic_fmc

```
axi_iic_fmc inst_axi_iic_fmc (
 s_axi_aclk(axi_aclk),
 s_axi_aresetn(axi_aresetn),
 iic2intc_irpt(iic2intc_irpt),
 s_axi_awaddr(iic_fmc_axi_awaddr[8:0]),
 s_axi_awvalid(iic_fmc_axi_awvalid),
 s_axi_awready(iic_fmc_axi_awready),
 s_axi_wdata(iic_fmc_axi_wdata),
 s_axi_wstrb(iic_fmc_axi_wstrb),
 s_axi_wvalid(iic_fmc_axi_wvalid),
 s_axi_wready(iic_fmc_axi_wready),
 s_axi_bresp(iic_fmc_axi_bresp),
```

```
                                                                                .
 s_axi_bvalid(iic_fmc_axi_bvalid),                                              .
 s_axi_bready(iic_fmc_axi_bready),                                              .
 s_axi_araddr(iic_fmc_axi_araddr[8:0]),                                         .
 s_axi_arvalid(iic_fmc_axi_arvalid),                                            .
 s_axi_arready(iic_fmc_axi_arready),                                            .
 s_axi_rdata(iic_fmc_axi_rdata),                                                .
 s_axi_rresp(iic_fmc_axi_rresp),                                                .
 s_axi_rvalid(iic_fmc_axi_rvalid),                                              .
 s_axi_rready(iic_fmc_axi_rready),                                              .
 sda_i(sda_i),                                                                  .
 sda_o(sda_o),                                                                  .
 sda_t(sda_t),                                                                  .
 scl_i(scl_i),                                                                  .
 scl_o(scl_o),                                                                  .
 scl_t(scl_t),                                                                  .
 gpo()
);
```

Module instance of axi_iic_fmc for the fmcomms5 device.

system_wrapper.v

AUTHORS

JAY CONVERTINO

DATES

2023/11/02

INFORMATION

Brief

System wrapper for pl and ps for zc702 board.

License MIT

system_wrapper

```
module system_wrapper #(
parameter
FPGA_TECHNOLOGY
 =
1,
parameter
FPGA_FAMILY
 =
4,
parameter
SPEED_GRADE
```

```
  =
20,
parameter
DEV_PACKAGE
  =
3,
parameter
DELAY_REFCLK_FREQUENCY
  =
200,
parameter
ADC_INIT_DELAY
  =
20,
parameter
DAC_INIT_DELAY
  =
0
) ( inout [14:0] ddr_addr, inout [ 2:0] ddr_ba, inout ddr_cas_n, inout ddr_c
```

System wrapper for pl and ps for zc702 board.

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. 10 is for -1.
DEV_PACKAGE parameter	Specify a number that is equal to the manufactures package. 14 is for cl.
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

Ports

ddr_addr	DDR interface
ddr_ba	DDR interface
ddr_cas_n	DDR interface
ddr_ck_n	DDR interface
ddr_ck_p	DDR interface
ddr_cke	DDR interface
ddr_cs_n	DDR interface
ddr_dm	DDR interface
ddr_dq	DDR interface
ddr_dqs_n	DDR interface
ddr_dqs_p	DDR interface
ddr_odt	DDR interface
ddr_ras_n	DDR interface

ddr_reset_n	DDR interface
ddr_we_n	DDR interface
fixed_io_ddr_vrn	DDR interface
fixed_io_ddr_vrp	DDR interface
fixed_io_mio	ps mio
fixed_io_ps_clk	ps clk
fixed_io_ps_porb	ps por
fixed_io_ps_srstb	ps rst
iic_scl_fmc	fmcomms5 i2c
iic_sda_fmc	fmcomms5 i2c
gpio_bd	gpio
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
gpio_status_0	fmcomms5 0 gpio
gpio_ctl_0	fmcomms5 0 gpio
gpio_en_agc_0	fmcomms5 0 gpio
gpio_resetb_0	fmcomms5 0 gpio
gpio_debug_1_0	fmcomms5 0 gpio
gpio_debug_2_0	fmcomms5 0 gpio
gpio_calsw_1_0	fmcomms5 0 gpio
gpio_calsw_2_0	fmcomms5 0 gpio
gpio_ad5355_rfen	fmcomms5 0 gpio
gpio_ad5355_lock	fmcomms5 0 gpio
txnrx_0	fmcomms5 0 txnrx
enable_0	fmcomms5 0 enable
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
gpio_status_1	fmcomms5 1 gpio
gpio_ctl_1	fmcomms5 1 gpio
gpio_en_agc_1	fmcomms5 1 gpio
gpio_resetb_1	fmcomms5 1 gpio
gpio_debug_1_1	fmcomms5 1 gpio
gpio_debug_2_1	fmcomms5 1 gpio
gpio_calsw_1_1	fmcomms5 1 gpio
gpio_calsw_2_1	fmcomms5 1 gpio
gpio_ad5355_rfen	fmcomms5 1 gpio
gpio_ad5355_lock	fmcomms5 1 gpio
txnrx_1	fmcomms5 1 txnrx
enable_1	fmcomms5 1 enable
mcs_sync	fmcomms5 sync
spi_ad9361_0	fmcomms5 ad9361 0 spi select
spi_ad9361_1	fmcomms5 ad9361 1 spi select
spi_ad5355	fmcomms5 ad5355 spi select
spi_clk	fmcomms5 spi clock
spi_mosi	fmcomms5 spi master out
spi_miso	fmcomms5 spi master in
ref_clk_p	fmcomms5 ref clock p
ref_clk_n	fmcomms5 ref clock n

INSTANTIANTED MODULES

i_ref_clk_ibuf

```
IBUFGDS i_ref_clk_ibuf (
.
I
(
ref_clk_p),
.
IB
(
ref_clk_n),
.
O
(
ref_clk_s)
)
```

Module instance of IBUFGDS for LVDS to cmos clock

i_ref_clk_rbuf

```
BUFR #(
.
BUFR_DIVIDE
("
BYPASS")
) i_ref_clk_rbuf ( .CLR (1'b0), .CE (1'b1), .I (ref_clk_s), .O (ref_clk))
```

Module instance of BUFR for cmos clock to clock region.

i_iobuf

```
ad_iobuf #(
.
DATA_WIDTH(42)
) i_iobuf ( .dio_t ({gpio_t[59:46], gpio_t[43:16]}), .dio_i ({gpio_o[59:46]
```

Module instance of ad_iobuf for tristate GPIO control.

i_gpio_bd

```
ad_iobuf #(
.
DATA_WIDTH(16)
) i_gpio_bd ( .dio_t (gpio_t[15:0]), .dio_i (gpio_o[15:0]), .dio_o (gpio_i[
```

Module instance of ad_iobuf for tristate GPIO bd control.

inst_system_pl_wrapper

```
system_pl_wrapper #(
.
FPGA_TECHNOLOGY(FPGA_TECHNOLOGY),
.
FPGA_FAMILY(FPGA_FAMILY),
.
SPEED_GRADE(SPEED_GRADE),
.
DEV_PACKAGE(DEV_PACKAGE),
.
ADC_INIT_DELAY(ADC_INIT_DELAY),
.
DAC_INIT_DELAY(DAC_INIT_DELAY),
.
DELAY_REFCLK_FREQUENCY(DELAY_REFCLK_FREQUENCY)
) inst_system_pl_wrapper ( .axi_aclk(s_axi_clk), .axi_aresetn(s_axi_aresetn
```

Module instance of system_pl_wrapper for the fmcomms5 device.

inst_system_ps_wrapper

```
system_ps_wrapper inst_system_ps_wrapper (
.GPIO_I(gpio_i),
.GPIO_O(gpio_o),
.GPIO_T(gpio_t),
.SPI0_SCLK_I(1'b0),
.SPI0_SCLK_O(spi_clk),
.SPI0_MOSI_I(1'b0),
.SPI0_MOSI_O(spi_mosi),
.SPI0_MISO_I(spi_miso),
.SPI0_SS_I(1'b1),
.SPI0_SS_O(spi_ad9361_0),
.SPI0_SS1_O(spi_ad9361_1),
.SPI0_SS2_O(spi_ad5355),
.SPI1_SCLK_I(1'b0),
.SPI1_SCLK_O(),
.SPI1_MOSI_I(1'b0),
.SPI1_MOSI_O(),
.SPI1_MISO_I(1'b0),
.SPI1_SS_I(1'b1),
.SPI1_SS_O(),
.SPI1_SS1_O(),
.SPI1_SS2_O(),
.M_AXI_araddr(w_axi_araddr),
.M_AXI_arprot(w_axi_arprot),
.M_AXI_arready(w_axi_arready),
.M_AXI_arvalid(w_axi_arvalid),
.M_AXI_awaddr(w_axi_awaddr),
.M_AXI_awprot(w_axi_awprot),
.M_AXI_awready(w_axi_awready),
.M_AXI_awvalid(w_axi_awvalid),
.M_AXI_bready(w_axi_bready),
.M_AXI_bresp(w_axi_bresp),
```

```
.M_AXI_bvalid(w_axi_bvalid),
.M_AXI_rdata(w_axi_rdata),
.M_AXI_rready(w_axi_rready),
.M_AXI_rresp(w_axi_rresp),
.M_AXI_rvalid(w_axi_rvalid),
.M_AXI_wdata(w_axi_wdata),
.M_AXI_wready(w_axi_wready),
.M_AXI_wstrb(w_axi_wstrb),
.M_AXI_wvalid(w_axi_wvalid),
.S_AXI_HP0_arready(),
.S_AXI_HP0_awready(adc_hp0_axi_awready),
.S_AXI_HP0_bvalid(adc_hp0_axi_bvalid),
.S_AXI_HP0_rlast(),
.S_AXI_HP0_rvalid(),
.S_AXI_HP0_wready(adc_hp0_axi_wready),
.S_AXI_HP0_bresp(adc_hp0_axi_bresp),
.S_AXI_HP0_rresp(),
.S_AXI_HP0_bid(),
.S_AXI_HP0_rid(),
.S_AXI_HP0_rdata(),
.S_AXI_HP0_ACLK(s_delay_clk),
.S_AXI_HP0_arvalid(1'b0),
.S_AXI_HP0_awvalid(adc_hp0_axi_awvalid),
.S_AXI_HP0_bready(adc_hp0_axi_bready),
.S_AXI_HP0_rready(1'b0),
.S_AXI_HP0_wlast(adc_hp0_axi_wlast),
.S_AXI_HP0_wvalid(adc_hp0_axi_wvalid),
.S_AXI_HP0_arburst(2'b01),
.S_AXI_HP0_arlock(0),
.S_AXI_HP0_arsize(3'b011),
.S_AXI_HP0_awburst(adc_hp0_axi_awburst),
.S_AXI_HP0_awlock(0),
.S_AXI_HP0_awsize(adc_hp0_axi_awsize),
```

```
.S_AXI_HP0_arprot(0),
.S_AXI_HP0_awprot(adc_hp0_axi_awprot),
.S_AXI_HP0_araddr(0),
.S_AXI_HP0_awaddr(adc_hp0_axi_awaddr),
.S_AXI_HP0_arcache(4'b0011),
.S_AXI_HP0_arlen(0),
.S_AXI_HP0_arqos(0),
.S_AXI_HP0_awcache(adc_hp0_axi_awcache),
.S_AXI_HP0_awlen(adc_hp0_axi_awlen),
.S_AXI_HP0_awqos(0),
.S_AXI_HP0_arid(0),
.S_AXI_HP0_awid(0),
.S_AXI_HP0_wid(0),
.S_AXI_HP0_wdata(adc_hp0_axi_wdata),
.S_AXI_HP0_wstrb(adc_hp0_axi_wstrb),
.S_AXI_HP1_arready(dac_hp1_axi_arready),
.S_AXI_HP1_awready(),
.S_AXI_HP1_bvalid(),
.S_AXI_HP1_rlast(dac_hp1_axi_rlast),
.S_AXI_HP1_rvalid(dac_hp1_axi_rvalid),
.S_AXI_HP1_wready(),
.S_AXI_HP1_bresp(),
.S_AXI_HP1_rresp(dac_hp1_axi_rresp),
.S_AXI_HP1_bid(),
.S_AXI_HP1_rid(),
.S_AXI_HP1_rdata(dac_hp1_axi_rdata),
.S_AXI_HP1_ACLK(s_delay_clk),
.S_AXI_HP1_arvalid(dac_hp1_axi_arvalid),
.S_AXI_HP1_awvalid(1'b0),
.S_AXI_HP1_bready(1'b0),
.S_AXI_HP1_rready(dac_hp1_axi_rready),
.S_AXI_HP1_wlast(1'b0),
.S_AXI_HP1_wvalid(1'b0),
```

```
.
S_AXI_HP1_arburst(dac_hp1_axi_arburst),
.
S_AXI_HP1_arlock(0),
.
S_AXI_HP1_arsize(dac_hp1_axi_arsize),
.
S_AXI_HP1_awburst(2'b01),
.
S_AXI_HP1_awlock(0),
.
S_AXI_HP1_awsize(3'b011),
.
S_AXI_HP1_arprot(dac_hp1_axi_arprot),
.
S_AXI_HP1_awprot(0),
.
S_AXI_HP1_araddr(dac_hp1_axi_araddr),
.
S_AXI_HP1_awaddr(0),
.
S_AXI_HP1_arcache(dac_hp1_axi_arcache),
.
S_AXI_HP1_arlen(dac_hp1_axi_arlen),
.
S_AXI_HP1_arqos(0),
.
S_AXI_HP1_awcache(4'b0011),
.
S_AXI_HP1_awlen(0),
.
S_AXI_HP1_awqos(0),
.
S_AXI_HP1_arid(0),
.
S_AXI_HP1_awid(0),
.
S_AXI_HP1_wid(0),
.
S_AXI_HP1_wdata(0),
.
S_AXI_HP1_wstrb(~0),
.
IRQ_F2P({{2{1'b0}}, s_adc_dma_irq, s_dac_dma_irq, s_iic2intc_irpt, {11{1'b0
.
FCLK_CLK0(s_axi_clk),
.
FCLK_CLK1(s_delay_clk),
.
FIXED_IO_mio(fixed_io_mio),
.
DDR_cas_n(ddr_cas_n),
.
DDR_cke(ddr_cke),
.
DDR_ck_n(ddr_ck_n),
.
DDR_ck_p(ddr_ck_p),
.
DDR_cs_n(ddr_cs_n),
.
DDR_reset_n(ddr_reset_n),
.
DDR_odt(ddr_odt),
.
DDR_ras_n(ddr_ras_n),
```

```
 DDR_we_n(ddr_we_n),
 DDR_ba(ddr_ba),
 DDR_addr(ddr_addr),
 FIXED_IO_ddr_vrn(fixed_io_ddr_vrn),
 FIXED_IO_ddr_vrp(fixed_io_ddr_vrp),
 DDR_dm(ddr_dm),
 DDR_dq(ddr_dq),
 DDR_dqs_n(ddr_dqs_n),
 DDR_dqs_p(ddr_dqs_p),
 FIXED_IO_ps_srstb(fixed_io_ps_srstb),
 FIXED_IO_ps_clk(fixed_io_ps_clk),
 FIXED_IO_ps_porb(fixed_io_ps_porb),
 peripheral_aresetn(s_axi_aresetn)
)
```

Module instance of inst_system_ps_wrapper for the built in CPU.

system_pl_wrapper.v

AUTHORS

JAY CONVERTINO

DATES

2023/11/02

INFORMATION

Brief

System wrapper for pl only for zc702 board.

License MIT

system_pl_wrapper

```
module system_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
) ( input axi_aclk, input axi_aresetn, input s_axi_awvalid, input [31:0] s_
```

System wrapper for pl only for zc702 board.

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. 10 is for -1.
DEV_PACKAGE parameter	Specify a number that is equal to the manufactures package. 14 is for cl.
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

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
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
iic_sda_fmc	i2c for fmc
iic_scl_fmc	i2c for fmc
iic2intc_irpt	i2c for fmc

INSTANTIANTED MODULES

iic_sda_iobuf

```
ad_iobuf #(
                                                                                .
DATA_WIDTH(1)
) iic_sda_iobuf ( .dio_t (sda_t), .dio_i (sda_o), .dio_o (sda_i), .dio_p (
```

Tristate i2c sda

iic_scl_iobuf

```
ad_iobuf #(
                                                                                .
DATA_WIDTH(1)
) iic_scl_iobuf ( .dio_t (scl_t), .dio_i (scl_o), .dio_o (scl_i), .dio_p (
```

Tristate i2c scl

inst_dma_rstgen

```
dma_rstgen inst_dma_rstgen (
                                                                                .
slowest_sync_clk(delay_clk),
                                                                                .
ext_reset_in(axi_aresetn),
                                                                                .
aux_reset_in(1'b1),
                                                                                .
mb_debug_sys_rst(1'b0),
                                                                                .
dcm_locked(1'b1),
                                                                                .
mb_reset(),
                                                                                .
bus_struct_reset(),
                                                                                .
peripheral_reset(),
                                                                                .
interconnect_aresetn(),
                                                                                .
peripheral_aresetn(m_axi_aresetn)
)
```

Generate a new DMA reset based on delay clock.

inst_ad9361x2_pl_wrapper

```
ad9361x2_pl_wrapper #(
                                                                                .
FPGA_TECHNOLOGY(FPGA_TECHNOLOGY),
                                                                                .
FPGA_FAMILY(FPGA_FAMILY),
```

```
                                                                                .
SPEED_GRADE(SPEED_GRADE),                                                       .
DEV_PACKAGE(DEV_PACKAGE),                                                       .
ADC_INIT_DELAY(ADC_INIT_DELAY),                                                 .
DAC_INIT_DELAY(DAC_INIT_DELAY),                                                 .
DELAY_REFCLK_FREQUENCY(DELAY_REFCLK_FREQUENCY)
) inst_ad9361x2_pl_wrapper ( .axi_aclk(axi_aclk), .axi_aresetn(axi_aresetn)
```

Module instance of inst_ad9361x2_pl_wrapper for the fmcomms5 device.

inst_axi_crossbar_pl

```
axi_crossbar_pl inst_axi_crossbar_pl (
                                                                                .
aclk(axi_aclk),                                                                 .
aresetn(axi_aresetn),                                                           .
s_axi_awaddr(s_axi_awaddr),                                                     .
s_axi_awprot(s_axi_awprot),                                                     .
s_axi_awvalid(s_axi_awvalid),                                                   .
s_axi_awready(s_axi_awready),                                                   .
s_axi_wdata(s_axi_wdata),                                                       .
s_axi_wstrb(s_axi_wstrb),                                                       .
s_axi_wvalid(s_axi_wvalid),                                                     .
s_axi_wready(s_axi_wready),                                                     .
s_axi_bresp(s_axi_bresp),                                                       .
s_axi_bvalid(s_axi_bvalid),                                                     .
s_axi_bready(s_axi_bready),                                                     .
s_axi_araddr(s_axi_araddr),                                                     .
s_axi_arprot(s_axi_arprot),                                                     .
s_axi_arvalid(s_axi_arvalid),                                                   .
s_axi_arready(s_axi_arready),                                                   .
s_axi_rdata(s_axi_rdata),                                                       .
s_axi_rresp(s_axi_rresp),                                                       .
s_axi_rvalid(s_axi_rvalid),                                                     .
s_axi_rready(s_axi_rready),                                                     .
m_axi_awaddr({iic_fmc_axi_awaddr, connect_axi_awaddr}),                         .
m_axi_awprot({iic_fmc_axi_awprot, connect_axi_awprot}),
```

```
 m_axi_awvalid({iic_fmc_axi_awvalid, connect_axi_awvalid}),
 m_axi_awready({iic_fmc_axi_awready, connect_axi_awready}),
 m_axi_wdata({iic_fmc_axi_wdata, connect_axi_wdata}),
 m_axi_wstrb({iic_fmc_axi_wstrb, connect_axi_wstrb}),
 m_axi_wvalid({iic_fmc_axi_wvalid, connect_axi_wvalid}),
 m_axi_wready({iic_fmc_axi_wready, connect_axi_wready}),
 m_axi_bresp({iic_fmc_axi_bresp, connect_axi_bresp}),
 m_axi_bvalid({iic_fmc_axi_bvalid, connect_axi_bvalid}),
 m_axi_bready({iic_fmc_axi_bready, connect_axi_bready}),
 m_axi_araddr({iic_fmc_axi_araddr, connect_axi_araddr}),
 m_axi_arprot({iic_fmc_axi_arprot, connect_axi_arprot}),
 m_axi_arvalid({iic_fmc_axi_arvalid, connect_axi_arvalid}),
 m_axi_arready({iic_fmc_axi_arready, connect_axi_arready}),
 m_axi_rdata({iic_fmc_axi_rdata, connect_axi_rdata}),
 m_axi_rresp({iic_fmc_axi_rresp, connect_axi_rresp}),
 m_axi_rvalid({iic_fmc_axi_rvalid, connect_axi_rvalid}),
 m_axi_rready({iic_fmc_axi_rready, connect_axi_rready})
)
```

Module instance of axi_crossbar_pl for the fmcomms5 device.

inst_axi_iic_fmc

```
axi_iic_fmc inst_axi_iic_fmc (
 s_axi_aclk(axi_aclk),
 s_axi_aresetn(axi_aresetn),
 iic2intc_irpt(iic2intc_irpt),
 s_axi_awaddr(iic_fmc_axi_awaddr[8:0]),
 s_axi_awvalid(iic_fmc_axi_awvalid),
 s_axi_awready(iic_fmc_axi_awready),
 s_axi_wdata(iic_fmc_axi_wdata),
 s_axi_wstrb(iic_fmc_axi_wstrb),
 s_axi_wvalid(iic_fmc_axi_wvalid),
 s_axi_wready(iic_fmc_axi_wready),
 s_axi_bresp(iic_fmc_axi_bresp),
```

```
                                                                                .
 s_axi_bvalid(iic_fmc_axi_bvalid),                                              .
 s_axi_bready(iic_fmc_axi_bready),                                              .
 s_axi_araddr(iic_fmc_axi_araddr[8:0]),                                         .
 s_axi_arvalid(iic_fmc_axi_arvalid),                                            .
 s_axi_arready(iic_fmc_axi_arready),                                            .
 s_axi_rdata(iic_fmc_axi_rdata),                                                .
 s_axi_rresp(iic_fmc_axi_rresp),                                                .
 s_axi_rvalid(iic_fmc_axi_rvalid),                                              .
 s_axi_rready(iic_fmc_axi_rready),                                              .
 sda_i(sda_i),                                                                  .
 sda_o(sda_o),                                                                  .
 sda_t(sda_t),                                                                  .
 scl_i(scl_i),                                                                  .
 scl_o(scl_o),                                                                  .
 scl_t(scl_t),                                                                  .
 gpo()
)
```

Module instance of axi_iic_fmc for the fmcomms5 device.

system_wrapper.v

AUTHORS

JAY CONVERTINO

DATES

2023/11/02

INFORMATION

Brief

System wrapper for pl and ps for zc706 board.

License MIT

system_wrapper

```
module system_wrapper #(
parameter
FPGA_TECHNOLOGY
 =
1,
parameter
FPGA_FAMILY
 =
4,
parameter
SPEED_GRADE
```

```
 =
20,
parameter
DEV_PACKAGE
 =
3,
parameter
DELAY_REFCLK_FREQUENCY
 =
200,
parameter
ADC_INIT_DELAY
 =
20,
parameter
DAC_INIT_DELAY
 =
0
) ( inout [14:0] ddr_addr, inout [ 2:0] ddr_ba, inout ddr_cas_n, inout ddr_c
```

System wrapper for pl and ps for zc706 board.

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. 10 is for -1.
DEV_PACKAGE parameter	Specify a number that is equal to the manufactures package. 14 is for cl.
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

Ports

ddr_addr	DDR interface
ddr_ba	DDR interface
ddr_cas_n	DDR interface
ddr_ck_n	DDR interface
ddr_ck_p	DDR interface
ddr_cke	DDR interface
ddr_cs_n	DDR interface
ddr_dm	DDR interface
ddr_dq	DDR interface
ddr_dqs_n	DDR interface
ddr_dqs_p	DDR interface
ddr_odt	DDR interface
ddr_ras_n	DDR interface

ddr_reset_n	DDR interface
ddr_we_n	DDR interface
fixed_io_ddr_vrn	DDR interface
fixed_io_ddr_vrp	DDR interface
fixed_io_mio	ps mio
fixed_io_ps_clk	ps clk
fixed_io_ps_porb	ps por
fixed_io_ps_srstb	ps rst
iic_scl_fmc	fmcomms5 i2c
iic_sda_fmc	fmcomms5 i2c
gpio_bd	gpio
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
gpio_status_0	fmcomms5 0 gpio
gpio_ctl_0	fmcomms5 0 gpio
gpio_en_agc_0	fmcomms5 0 gpio
gpio_resetb_0	fmcomms5 0 gpio
gpio_debug_1_0	fmcomms5 0 gpio
gpio_debug_2_0	fmcomms5 0 gpio
gpio_calsw_1_0	fmcomms5 0 gpio
gpio_calsw_2_0	fmcomms5 0 gpio
gpio_ad5355_rfen	fmcomms5 0 gpio
gpio_ad5355_lock	fmcomms5 0 gpio
txnrx_0	fmcomms5 0 txnrx
enable_0	fmcomms5 0 enable
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
gpio_status_1	fmcomms5 1 gpio
gpio_ctl_1	fmcomms5 1 gpio
gpio_en_agc_1	fmcomms5 1 gpio
gpio_resetb_1	fmcomms5 1 gpio
gpio_debug_1_1	fmcomms5 1 gpio
gpio_debug_2_1	fmcomms5 1 gpio
gpio_calsw_1_1	fmcomms5 1 gpio
gpio_calsw_2_1	fmcomms5 1 gpio
gpio_ad5355_rfen	fmcomms5 1 gpio
gpio_ad5355_lock	fmcomms5 1 gpio
txnrx_1	fmcomms5 1 txnrx
enable_1	fmcomms5 1 enable
mcs_sync	fmcomms5 sync
spi_ad9361_0	fmcomms5 ad9361 0 spi select
spi_ad9361_1	fmcomms5 ad9361 1 spi select
spi_ad5355	fmcomms5 ad5355 spi select
spi_clk	fmcomms5 spi clock
spi_mosi	fmcomms5 spi master out
spi_miso	fmcomms5 spi master in
ref_clk_p	fmcomms5 ref clock p
ref_clk_n	fmcomms5 ref clock n

INSTANTIANTED MODULES

i_ref_clk_ibuf

```
IBUFGDS i_ref_clk_ibuf (
.
I
(
ref_clk_p),
.
IB
(
ref_clk_n),
.
O
(
ref_clk_s)
)
```

Module instance of IBUFGDS for LVDS to cmos clock

i_ref_clk_rbuf

```
BUFR #(
.
BUFR_DIVIDE
("
BYPASS")
) i_ref_clk_rbuf ( .CLR (1'b0), .CE (1'b1), .I (ref_clk_s), .O (ref_clk))
```

Module instance of BUFR for cmos clock to clock region.

i_iobuf

```
ad_iobuf #(
.
DATA_WIDTH(57)
) i_iobuf ( .dio_t ({gpio_t[59:46], gpio_t[43:16], gpio_t[14:0]}), .dio_i (
```

Module instance of ad_iobuf for tristate GPIO control.

inst_system_pl_wrapper

```
system_pl_wrapper #(
.
FPGA_TECHNOLOGY(FPGA_TECHNOLOGY),
.
FPGA_FAMILY(FPGA_FAMILY),
.
SPEED_GRADE(SPEED_GRADE),
.
DEV_PACKAGE(DEV_PACKAGE),
.
ADC_INIT_DELAY(ADC_INIT_DELAY),
.
DAC_INIT_DELAY(DAC_INIT_DELAY),
.
DELAY_REFCLK_FREQUENCY(DELAY_REFCLK_FREQUENCY)
) inst_system_pl_wrapper ( .axi_aclk(s_axi_clk), .axi_aresetn(s_axi_aresetn
```

Module instance of system_pl_wrapper for the fmcomms5 device.

inst_system_ps_wrapper

```
system_ps_wrapper inst_system_ps_wrapper (
.
GPIO_I(gpio_i),
.
GPIO_O(gpio_o),
.
GPIO_T(gpio_t),
.
SPI0_SCLK_I(1'b0),
```

```
    SPI0_SCLK_O(spi_clk),
    SPI0_MOSI_I(1'b0),
    SPI0_MOSI_O(spi_mosi),
    SPI0_MISO_I(spi_miso),
    SPI0_SS_I(1'b1),
    SPI0_SS_O(spi_ad9361_0),
    SPI0_SS1_O(spi_ad9361_1),
    SPI0_SS2_O(spi_ad5355),
    SPI1_SCLK_I(1'b0),
    SPI1_SCLK_O(),
    SPI1_MOSI_I(1'b0),
    SPI1_MOSI_O(),
    SPI1_MISO_I(1'b0),
    SPI1_SS_I(1'b1),
    SPI1_SS_O(),
    SPI1_SS1_O(),
    SPI1_SS2_O(),
    M_AXI_araddr(w_axi_araddr),
    M_AXI_arprot(w_axi_arprot),
    M_AXI_arready(w_axi_arready),
    M_AXI_arvalid(w_axi_arvalid),
    M_AXI_awaddr(w_axi_awaddr),
    M_AXI_awprot(w_axi_awprot),
    M_AXI_awready(w_axi_awready),
    M_AXI_awvalid(w_axi_awvalid),
    M_AXI_bready(w_axi_bready),
    M_AXI_bresp(w_axi_bresp),
    M_AXI_bvalid(w_axi_bvalid),
    M_AXI_rdata(w_axi_rdata),
    M_AXI_rready(w_axi_rready),
    M_AXI_rresp(w_axi_rresp),
    M_AXI_rvalid(w_axi_rvalid),
    M_AXI_wdata(w_axi_wdata),
```

```
M_AXI_wready(w_axi_wready),
M_AXI_wstrb(w_axi_wstrb),
M_AXI_wvalid(w_axi_wvalid),
S_AXI_HP0_arready(),
S_AXI_HP0_awready(adc_hp0_axi_awready),
S_AXI_HP0_bvalid(adc_hp0_axi_bvalid),
S_AXI_HP0_rlast(),
S_AXI_HP0_rvalid(),
S_AXI_HP0_wready(adc_hp0_axi_wready),
S_AXI_HP0_bresp(adc_hp0_axi_bresp),
S_AXI_HP0_rresp(),
S_AXI_HP0_bid(),
S_AXI_HP0_rid(),
S_AXI_HP0_rdata(),
S_AXI_HP0_ACLK(s_delay_clk),
S_AXI_HP0_arvalid(1'b0),
S_AXI_HP0_awvalid(adc_hp0_axi_awvalid),
S_AXI_HP0_bready(adc_hp0_axi_bready),
S_AXI_HP0_rready(1'b0),
S_AXI_HP0_wlast(adc_hp0_axi_wlast),
S_AXI_HP0_wvalid(adc_hp0_axi_wvalid),
S_AXI_HP0_arburst(2'b01),
S_AXI_HP0_arlock(0),
S_AXI_HP0_arsize(3'b011),
S_AXI_HP0_awburst(adc_hp0_axi_awburst),
S_AXI_HP0_awlock(0),
S_AXI_HP0_awsize(adc_hp0_axi_awsize),
S_AXI_HP0_arprot(0),
S_AXI_HP0_awprot(adc_hp0_axi_awprot),
S_AXI_HP0_araddr(0),
S_AXI_HP0_awaddr(adc_hp0_axi_awaddr),
S_AXI_HP0_arcache(4'b0011),
S_AXI_HP0_arlen(0),
```

```
.
S_AXI_HP0_arqos(0),
.
S_AXI_HP0_awcache(adc_hp0_axi_awcache),
.
S_AXI_HP0_awlen(adc_hp0_axi_awlen),
.
S_AXI_HP0_awqos(0),
.
S_AXI_HP0_arid(0),
.
S_AXI_HP0_awid(0),
.
S_AXI_HP0_wid(0),
.
S_AXI_HP0_wdata(adc_hp0_axi_wdata),
.
S_AXI_HP0_wstrb(adc_hp0_axi_wstrb),
.
S_AXI_HP1_arready(dac_hp1_axi_arready),
.
S_AXI_HP1_awready(),
.
S_AXI_HP1_bvalid(),
.
S_AXI_HP1_rlast(dac_hp1_axi_rlast),
.
S_AXI_HP1_rvalid(dac_hp1_axi_rvalid),
.
S_AXI_HP1_wready(),
.
S_AXI_HP1_bresp(),
.
S_AXI_HP1_rresp(dac_hp1_axi_rresp),
.
S_AXI_HP1_bid(),
.
S_AXI_HP1_rid(),
.
S_AXI_HP1_rdata(dac_hp1_axi_rdata),
.
S_AXI_HP1_ACLK(s_delay_clk),
.
S_AXI_HP1_arvalid(dac_hp1_axi_arvalid),
.
S_AXI_HP1_awvalid(1'b0),
.
S_AXI_HP1_bready(1'b0),
.
S_AXI_HP1_rready(dac_hp1_axi_rready),
.
S_AXI_HP1_wlast(1'b0),
.
S_AXI_HP1_wvalid(1'b0),
.
S_AXI_HP1_arburst(dac_hp1_axi_arburst),
.
S_AXI_HP1_arlock(0),
.
S_AXI_HP1_arsize(dac_hp1_axi_arsize),
.
S_AXI_HP1_awburst(2'b01),
.
S_AXI_HP1_awlock(0),
.
S_AXI_HP1_awsize(3'b011),
```

```
.
S_AXI_HP1_arprot(dac_hp1_axi_arprot),
.
S_AXI_HP1_awprot(0),
.
S_AXI_HP1_araddr(dac_hp1_axi_araddr),
.
S_AXI_HP1_awaddr(0),
.
S_AXI_HP1_arcache(dac_hp1_axi_arcache),
.
S_AXI_HP1_arlen(dac_hp1_axi_arlen),
.
S_AXI_HP1_arqos(0),
.
S_AXI_HP1_awcache(4'b0011),
.
S_AXI_HP1_awlen(0),
.
S_AXI_HP1_awqos(0),
.
S_AXI_HP1_arid(0),
.
S_AXI_HP1_awid(0),
.
S_AXI_HP1_wid(0),
.
S_AXI_HP1_wdata(0),
.
S_AXI_HP1_wstrb(~0),
.
IRQ_F2P({{2{1'b0}}, s_adc_dma_irq, s_dac_dma_irq, s_iic2intc_irpt, {11{1'b0
.
FCLK_CLK0(s_axi_clk),
.
FCLK_CLK1(s_delay_clk),
.
FIXED_IO_mio(fixed_io_mio),
.
DDR_cas_n(ddr_cas_n),
.
DDR_cke(ddr_cke),
.
DDR_ck_n(ddr_ck_n),
.
DDR_ck_p(ddr_ck_p),
.
DDR_cs_n(ddr_cs_n),
.
DDR_reset_n(ddr_reset_n),
.
DDR_odt(ddr_odt),
.
DDR_ras_n(ddr_ras_n),
.
DDR_we_n(ddr_we_n),
.
DDR_ba(ddr_ba),
.
DDR_addr(ddr_addr),
.
FIXED_IO_ddr_vrn(fixed_io_ddr_vrn),
.
FIXED_IO_ddr_vrp(fixed_io_ddr_vrp),
.
DDR_dm(ddr_dm),
```

```
 DDR_dq(ddr_dq),                                                                 .
 DDR_dqs_n(ddr_dqs_n),                                                           .
 DDR_dqs_p(ddr_dqs_p),                                                           .
 FIXED_IO_ps_srstb(fixed_io_ps_srstb),                                           .
 FIXED_IO_ps_clk(fixed_io_ps_clk),                                               .
 FIXED_IO_ps_porb(fixed_io_ps_porb),                                             .
 peripheral_aresetn(s_axi_aresetn)
)
```

Module instance of inst_system_ps_wrapper for the built in CPU.

system_pl_wrapper.v

AUTHORS

JAY CONVERTINO

DATES

2023/11/02

INFORMATION

Brief

System wrapper for pl only for zcu102 board.

License MIT

system_pl_wrapper

```
module system_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
) ( input axi_aclk, input axi_aresetn, input s_axi_awvalid, input [31:0] s_
```

System wrapper for pl only for zcu102 board.

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. 10 is for -1.
DEV_PACKAGE parameter	Specify a number that is equal to the manufactures package. 14 is for cl.
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

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
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_dma_rstgen

```
dma_rstgen inst_dma_rstgen (
                                                                                .
slowest_sync_clk(m_axi_aclk),
                                                                                .
ext_reset_in(axi_aresetn),
                                                                                .
aux_reset_in(1'b1),
                                                                                .
mb_debug_sys_rst(1'b0),
                                                                                .
dcm_locked(1'b1),
                                                                                .
mb_reset(),
                                                                                .
bus_struct_reset(),
                                                                                .
peripheral_reset(),
                                                                                .
interconnect_aresetn(),
                                                                                .
peripheral_aresetn(m_axi_aresetn)
)
```

Generate a new DMA reset based on delay clock.

system_wrapper.v

AUTHORS

JAY CONVERTINO

DATES

2023/11/02

INFORMATION

Brief

System wrapper for pl and ps for zcu102 board.

License MIT

system_wrapper

```
module system_wrapper #(
parameter
FPGA_TECHNOLOGY
 =
3,
parameter
FPGA_FAMILY
 =
4,
parameter
SPEED_GRADE
```

```
 =
20,
parameter
DEV_PACKAGE
 =
3,
parameter
DELAY_REFCLK_FREQUENCY
 =
500,
parameter
ADC_INIT_DELAY
 =
8,
parameter
DAC_INIT_DELAY
 =
0
) ( input [12:0] gpio_bd_i, output [ 7:0] gpio_bd_o, input rx_clk_in_0_p, ir
```

System wrapper for pl and ps for zcu102 board.

Parameters

FPGA_TECHNOLOGY parameter	Type of FPGA, such as Ultrascale, Arria 10. 3 is for ultrascale+.
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

Ports

gpio_bd_i	gpio
gpio_bd_o	gpio
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
gpio_status_0	fmcomms5 0 gpio
gpio_ctl_0	fmcomms5 0 gpio
gpio_en_agc_0	fmcomms5 0 gpio
gpio_resetb_0	fmcomms5 0 gpio
gpio_debug_1_0	fmcomms5 0 gpio
gpio_debug_2_0	fmcomms5 0 gpio
gpio_calsw_1_0	fmcomms5 0 gpio
gpio_calsw_2_0	fmcomms5 0 gpio
gpio_ad5355_rfen	fmcomms5 0 gpio
gpio_ad5355_lock	fmcomms5 0 gpio
txnrx_0	fmcomms5 0 txnrx
enable_0	fmcomms5 0 enable
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
gpio_status_1	fmcomms5 1 gpio
gpio_ctl_1	fmcomms5 1 gpio
gpio_en_agc_1	fmcomms5 1 gpio
gpio_resetb_1	fmcomms5 1 gpio
gpio_debug_1_1	fmcomms5 1 gpio
gpio_debug_2_1	fmcomms5 1 gpio
gpio_calsw_1_1	fmcomms5 1 gpio
gpio_calsw_2_1	fmcomms5 1 gpio
txnrx_1	fmcomms5 1 txnrx
enable_1	fmcomms5 1 enable
mcs_sync	fmcomms5 sync
spi_ad9361_0	fmcomms5 ad9361 0 spi select
spi_ad9361_1	fmcomms5 ad9361 1 spi select
spi_ad5355	fmcomms5 ad5355 spi select
spi_clk	fmcomms5 spi clock
spi_mosi	fmcomms5 spi master out

spi_miso	fmcomms5 spi master in
ref_clk_p	fmcomms5 ref clock p
ref_clk_n	fmcomms5 ref clock n

INSTANTIANTED MODULES

i_ref_clk_ibuf_ds

```
IBUFDS i_ref_clk_ibuf_ds (
.
I
(
ref_clk_p),
.
IB
(
ref_clk_n),
.
O
(
ref_clk_s_ds)
)
```

Module instance of IBUFGDS for LVDS to cmos clock

i_ref_clk_ibuf

```
BUFG i_ref_clk_ibuf (
.
I
(
ref_clk_s_ds),
.
O
(
ref_clk_s)
)
```

Module instance of BUFG for cmos clock

i_ref_clk_rbuf

```
BUFR #(
.
BUFR_DIVIDE
("
BYPASS")
) i_ref_clk_rbuf ( .CLR (1'b0), .CE (1'b1), .I (ref_clk_s), .O (ref_clk))
```

Module instance of BUFR for cmos clock

inst_system_pl_wrapper

```
system_pl_wrapper #(
                                                                                .
FPGA_TECHNOLOGY(FPGA_TECHNOLOGY),
                                                                                .
FPGA_FAMILY(FPGA_FAMILY),
                                                                                .
SPEED_GRADE(SPEED_GRADE),
                                                                                .
DEV_PACKAGE(DEV_PACKAGE),
                                                                                .
ADC_INIT_DELAY(ADC_INIT_DELAY),
                                                                                .
DAC_INIT_DELAY(DAC_INIT_DELAY),
                                                                                .
DELAY_REFCLK_FREQUENCY(DELAY_REFCLK_FREQUENCY)
) inst_system_pl_wrapper ( .axi_aclk(s_axi_clk), .axi_aresetn(s_axi_aresetn
```

Module instance of system_pl_wrapper for the fmcomms2-3 device.