Multi-Instance Remote RFDC Reference Design

User Guide

UG1552 (v1.0) March 16, 2022

Xilinx is creating an environment where employees, customers, and partners feel welcome and included. To that end, we’re removing non-inclusive language from our products and related collateral. We’ve launched an internal initiative to remove language that could exclude people or reinforce historical biases, including terms embedded in our software and IPs. You may still find examples of non-inclusive language in our older products as we work to make these changes and align with evolving industry standards. Follow this link for more information.

Table of Contents

Chapter 1

Introduction

The objective of this reference design is to help you quickly and easily evaluate bridging the RF data converter between systems using the Advanced eXtensible Interface (AXI) for two device system-on-chip solutions. This application note demonstrates how to access the RFDC on remote secondary systems using the AXI interface. The Xilinx® LogiCORE™ IP AXI Chip2Chip core provides bridging between systems by means of SFP connectors.

Chapter 2

Included Systems

The reference design includes Vivado® 2021.1 primary and remote RFSoC devices. Vivado Design Suite helps simplify the task of instantiating, configuring, and connecting IP blocks to form complete integrated systems.

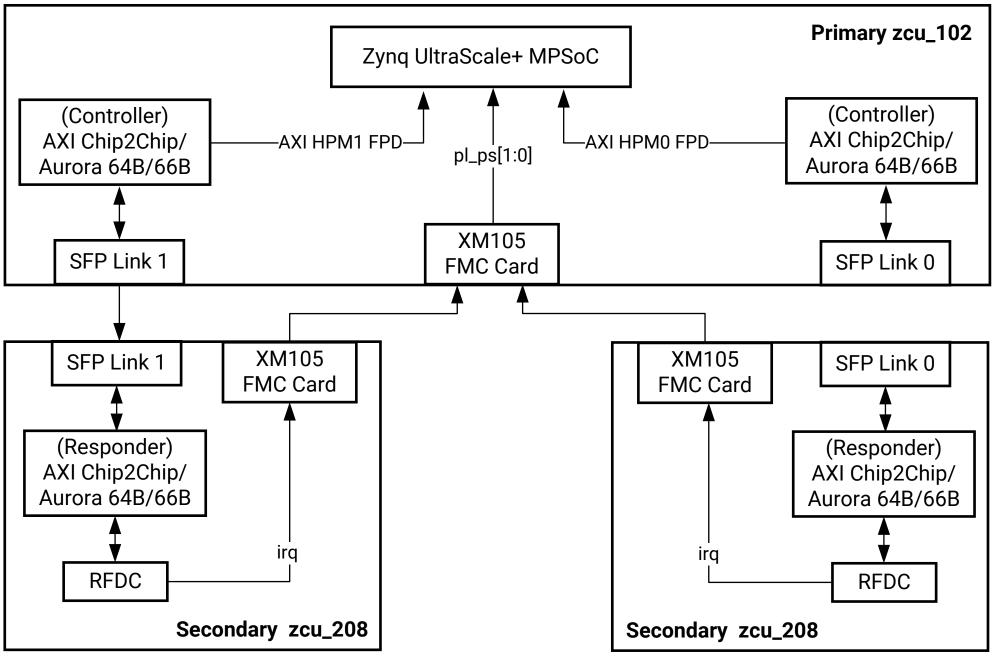
The reference design also includes a PetaLinux 2021.1 project with the RFDC driver patch required to access the remote RFSoC devices from the primary control board (ZCU102). Software configuration tools are available to extract the secondary design details from the RFDC, these configuration parameters are added to the custom RFDC dtsi (device tree include). Sample Linux applications are provided to demonstrate read/write access to the secondary RFDC and provide interrupt processing to read error registers.

Chapter 3

System Overview

The primary controller communicates with two secondary controllers, as shown in the following block diagram.

Figure 1: **Primary and Secondary Design Block Diagram**



X26376-030822

Hardware Requirements

Hardware requirements are as follows:

- ZCU102 evaluation board (primary controller)
    - XM105 HW FMC debug card
    - SFP PL AXI Ethernet plugs
    - Power supply and cables
- ZCU208 evaluation board (secondary controller #1)
    - XM105 HW FMC debug card
    - SFP PL AXI Ethernet plugs
    - Power supply and cables
- ZCU208 evaluation board (secondary controller #2)
    - XM105 HW FMC debug card
    - SFP PL AXI Ethernet plugs
    - Power supply and cables

The two RFSoC platforms are identical in design and implementation. Both provide an AXI interface to the RFDC and interrupt signals from each are routed back to the primary ZCU102.

Chapter 5

Core IP Overview

LogiCORE™ IP provides bridge communications to enable off-chip AXI communication from the primary ZCU102 to the secondary ZCU208 RFDC.

RFSoC RF Data Converter v2.5

The Xilinx® Zynq® UltraScale+™ RFSoC family integrates the key subsystems required to implement a complete software-defined radio including direct RF sampling data converters, enabling eCPRI and Gigabit Ethernet-to-RF on a single, highly programmable SoC. Each RFSoC offers multiple RF-sampling analog-to-digital (RF-ADC) and RF-sampling digital-to-analog (RF-DAC) data converters. The data converters are high-precision, high-speed, and power efficient. Both are highly configurable and tightly integrated with the programmable logic (PL) resources of the Zynq UltraScale+ RFSoC. The RF-ADC supports device-dependent sample rates and input signal frequencies listed in the *Zynq UltraScale+ RFSoC Data Sheet: Overview* (DS889), with excellent dynamic range performance. The RF-DAC generates output carrier frequencies at rates defined in the *Zynq UltraScale+ RFSoC Data Sheet: DC and AC Switching Characteristics* (DS926) depending on the device (see DS889 for device information).

The RF data converters also include power-efficient digital down converters (DDCs) and digital up converters (DUCs) that include programmable interpolation and decimation rates, a numerically controlled oscillator (NCO), and a complex mixer. The DDCs and DUCs can also support multiband operation. The following figure shows the block diagram of the Zynq UltraScale+ RFSoC RF data converter. The RF-ADCs and RF-DACs are organized into tiles, each containing one, two, or four RF-ADCs or one, two, or four RF-DACs. Multiple tiles are available in each Zynq UltraScale+ RFSoC (see the specific device data sheet for the number of tiles and converters per device). Each tile also includes a block with a PLL and all the necessary clock handling logic and distribution routing for the analog and digital logic

AXI Chip2Chip

The LogiCORE™ IP AXI Chip2Chip is a soft Xilinx IP core for use with Vivado Design Suite. The adaptable block provides bridging between AXI systems for multi-device system on-chip (SoC) solutions. The core supports multiple device-to-device interfacing options and provides a low pin count, high performance AXI chip-to-chip bridging solution.

Aurora 64B/66B

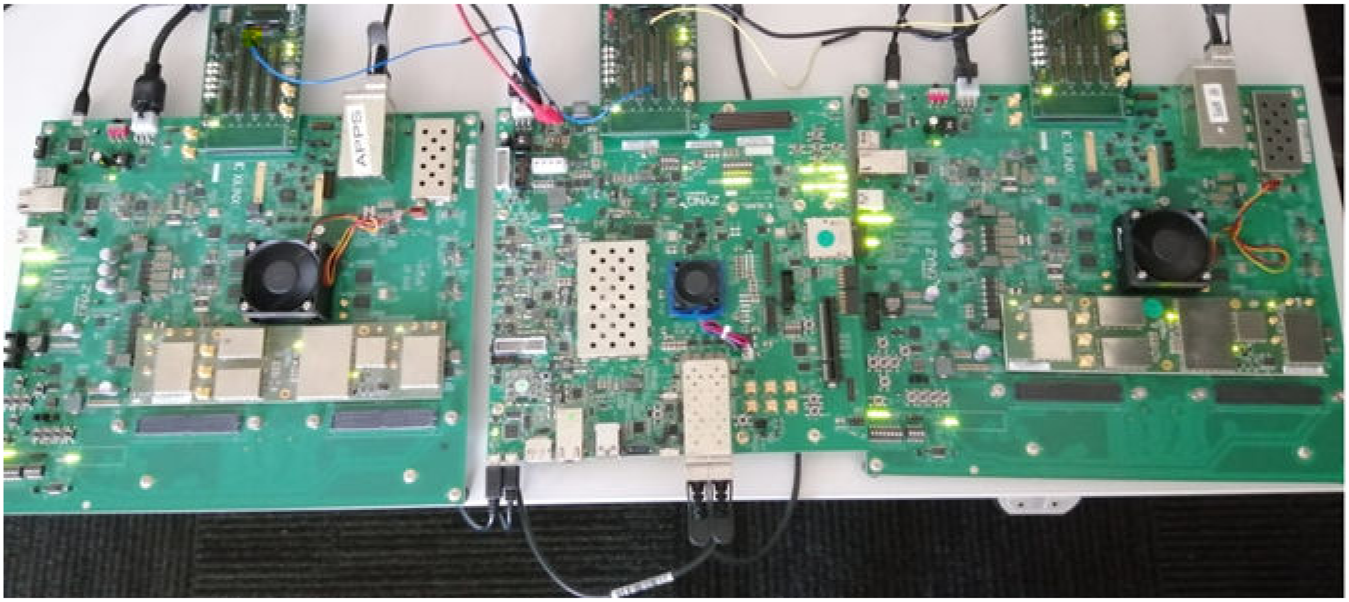
Aurora is a LogiCORE™ IP designed to enable easy implementation of Xilinx transceivers while providing a light-weight user interface on top of which designers can build a serial link. Aurora 64B/66B is a scalable, lightweight, link-layer protocol for high-speed serial communication. The protocol specification is open and available upon request. The IP is free to use from the IP Catalog on Xilinx® silicon devices.

Aurora is typically used in applications requiring low-cost, high data-rate, scalable and flexible means to build a serial data channel. Its simple framing structure can easily be used to encapsulate data from existing protocols, and its electrical requirements are compatible with commodity equipment. Aurora can be used to provide increased performance without high FPGA resource costs, software redevelopment, or exotic physical infrastructure.

Hardware Implementation

The following figure showd a completely connected system. Descriptions of the primary controller, secondary controller, and FMC debug card are provided in the following sections.

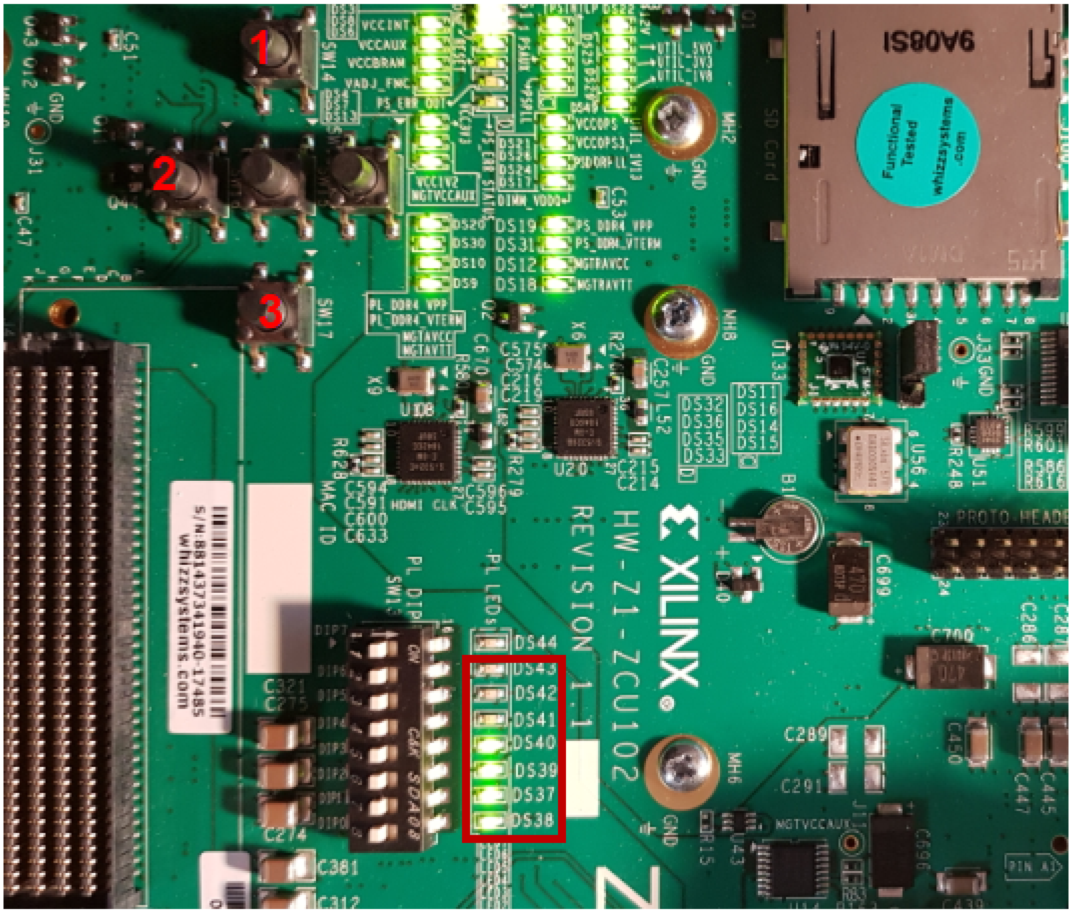
Figure 2: **Hardware Implementation**



Primary Controller

The primary controller board is shown in the following figure. The status LEDs and push-button switches 1,2, and 3 are highlighted.

Figure 3: **Primary Controller (ZCU102)**



Status LEDs

Table 1: **Status LED Identification**

LED	Signal	Link	Working Status
DS43	Link Status	1	ON
DS42	Channel up	1	ON
DS41	Lane up	1	ON
DS40	Link status	0	ON
DS39	Channel up	0	ON
DS37	Lane up	0	ON
DS38	PS clock	-	Blinking

Switches

Push-button switches used to reset links are identified in the following table.

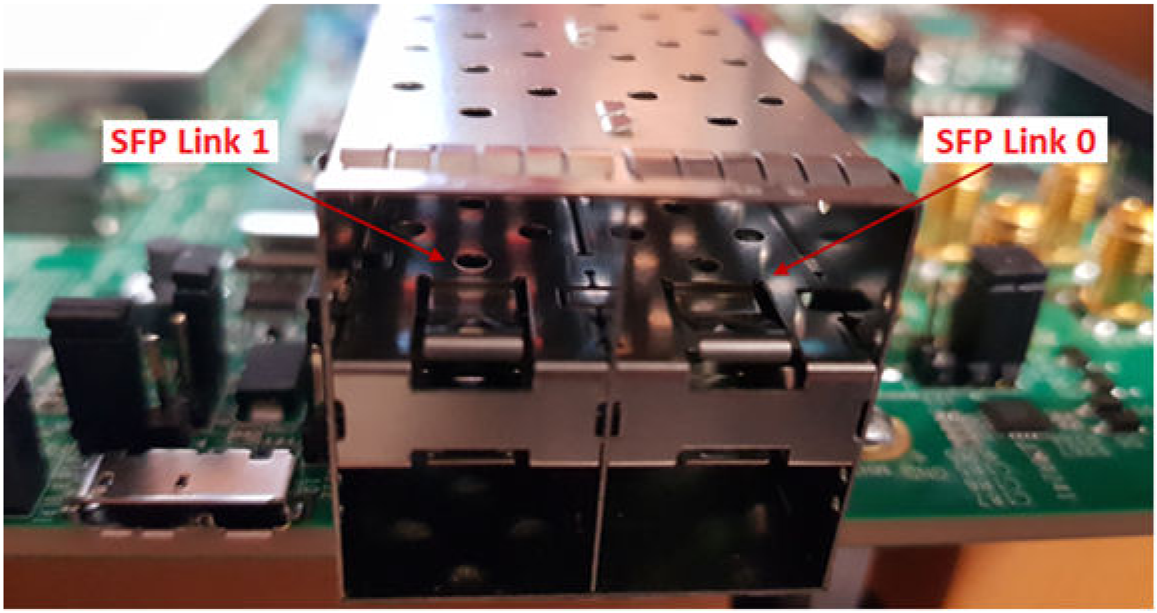
Table 2: **Links Reset**

Switch	Link
Push-button 1	Core Reset Link 1 and Link 2
Push-button 2	PMA Reset Link 1 (Transceivers and Core)
Push-button 3	PMA Reset Link 0 (Transceivers and Core)

SFP Connection

The SFP connection is shown in the following figure.

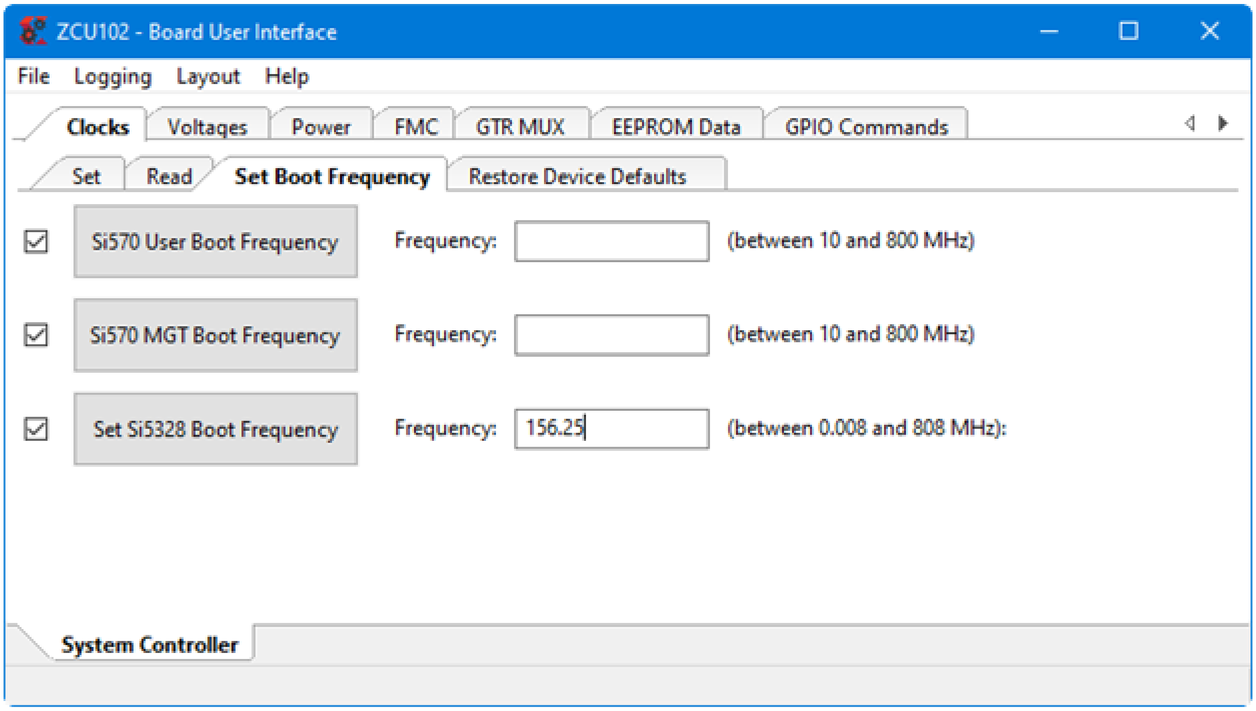
Figure 4: **SFP Connection**



Transceiver Reference Clock Setup

The PLL that generates the reference clock for the transceiver must be programmed to generate 156.25 MHz. Files are provided with the tool. The following screenshot shows the setup to program the PLL Si5328. By using Set Boot frequency, the clock can be programmed once and there is no need to reprogram it after a power cycle.

Figure 5: **PLL Programming**



☑ Si570 User Boot Frequency Frequency: (between 10 and 800 MHz)

☑ Si570 MGT Boot Frequency Frequency: (between 10 and 800 MHz)

☑ Set Si5328 Boot Frequency Frequency: 156.25 (between 0.008 and 808 MHz):

Primary PS-PL AXI Interfaces

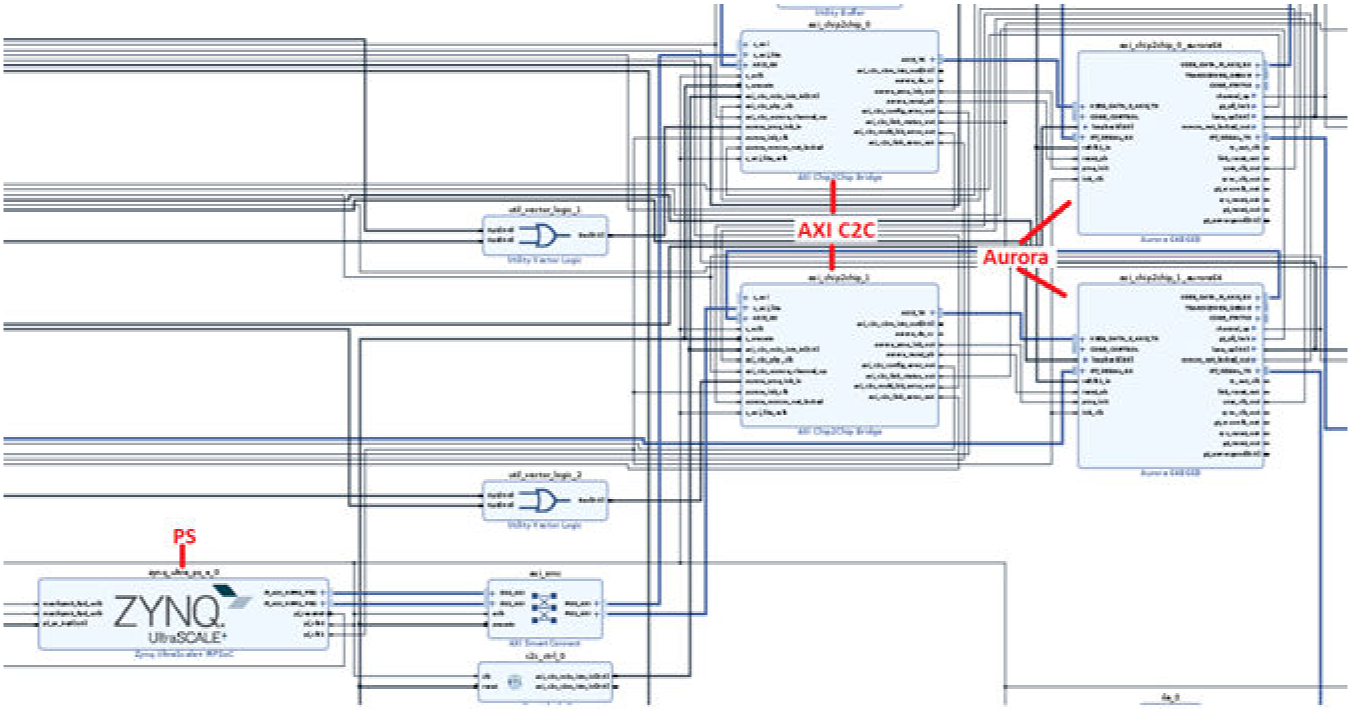
The core mechanism for cross-component communication in the Zynq UltraScale+ MPSoC device is the ARM AXI interface. For this design, two general-purpose AXI interfaces are used:

Table 3: **AXI Interfaces**

AXI Interface	Description	Address
AXI HPM0 FPD	High performance master 0 in full power mode.	0xa0000000
AXI HPM1 FPD	High performance master 1 in full power mode.	0xa0040000

The two AXI interfaces are connected to the SFP connectors, allowing secondary device connection to the RFDC IP. The primary chip2chip connection is shown in the following figure.

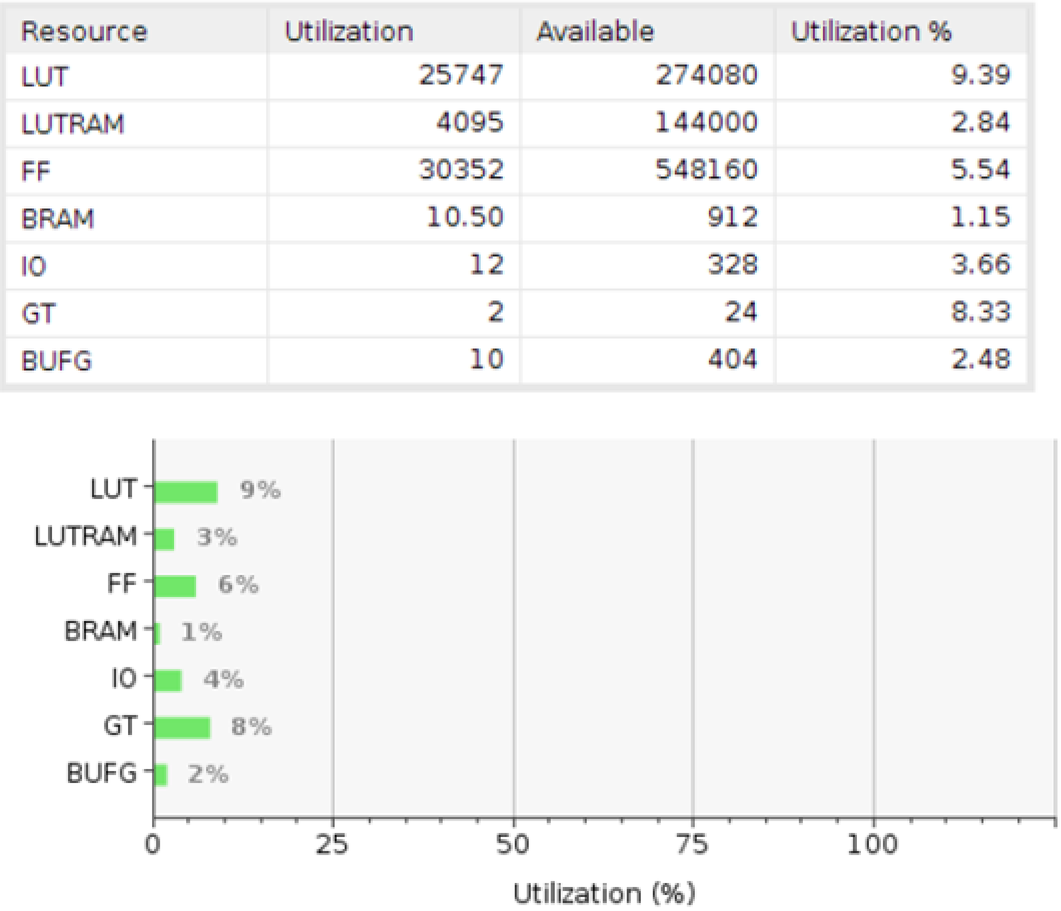
Figure 6: **Primary Chip2Chip Connection**



Utilization and Performance

Resources utilization for the secondary reference design is provided in the following figure.

Figure 7: **Utilization and Performance**

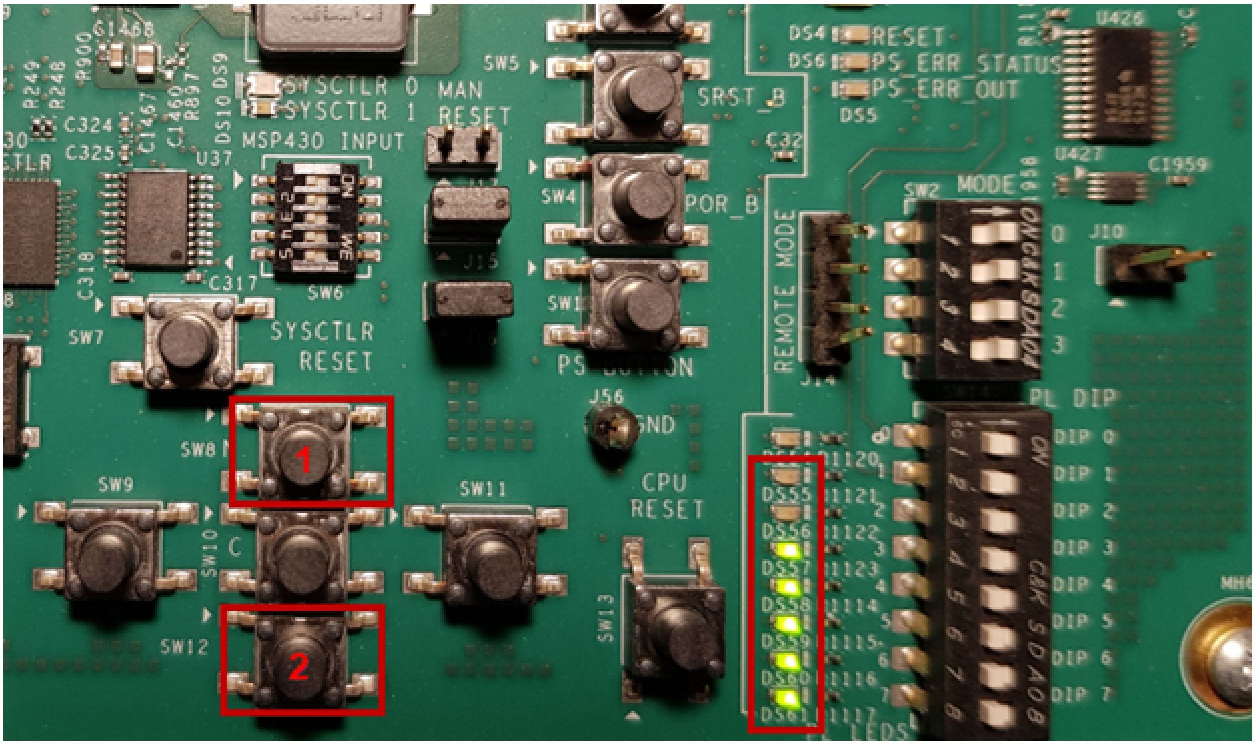


Resource	Utilization	Available	Utilization %
LUT	25747	274080	9.39
LUTRAM	4095	144000	2.84
FF	30352	548160	5.54
BRAM	10.50	912	1.15
IO	12	328	3.66
GT	2	24	8.33
BUFG	10	404	2.48

Secondary Controller

The secondary controller board is shown in the following figure. The status LEDs and push-button switches 1 and 2 are highlighted.

Figure 8: **Secondary Controller (ZCU208)**



Status LEDs

Table 4: **Status LED Identification**

LED	Signal	Working Status
DS555	Multi_bit_error	OFF
DS556	Config_error	OFF
DS557	Link_status	ON
DS558	Channel up	ON
DS559	Lane up	ON
DS560	GT PLL Lock	ON
DS561	PS Clock	Blinking

Switches

Push-button switches used to reset links are identified in the following table.

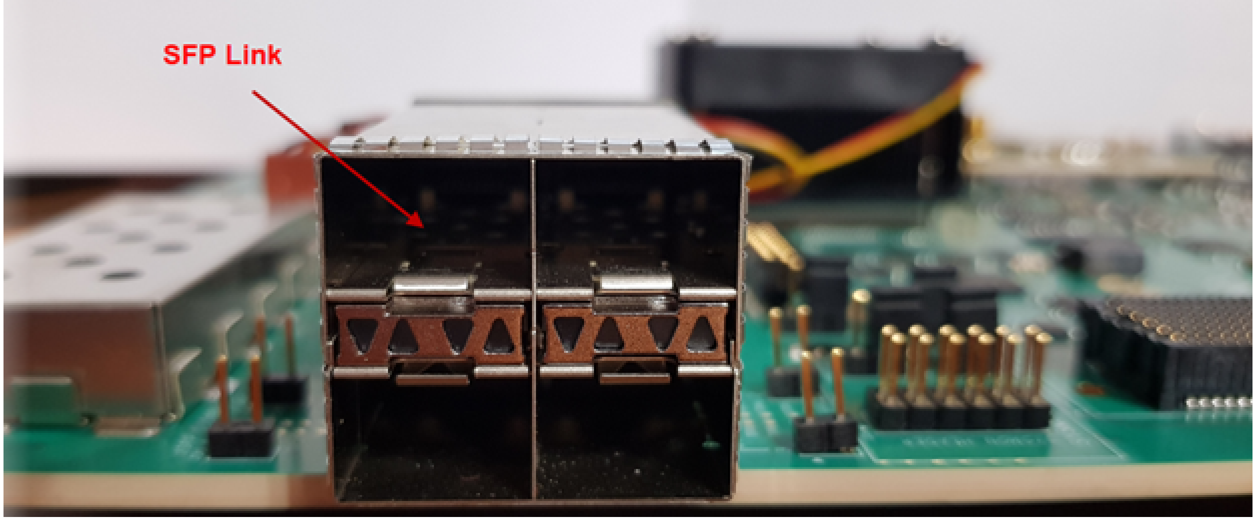
Table 5: **Links Reset**

Switch	Link
Push-button 1	PMA Reset (Transceivers and Core)
Push-button 2	Core Reset

SFP Connection

The SFP connection is shown in the following figure.

Figure 9: **SFP Connection**

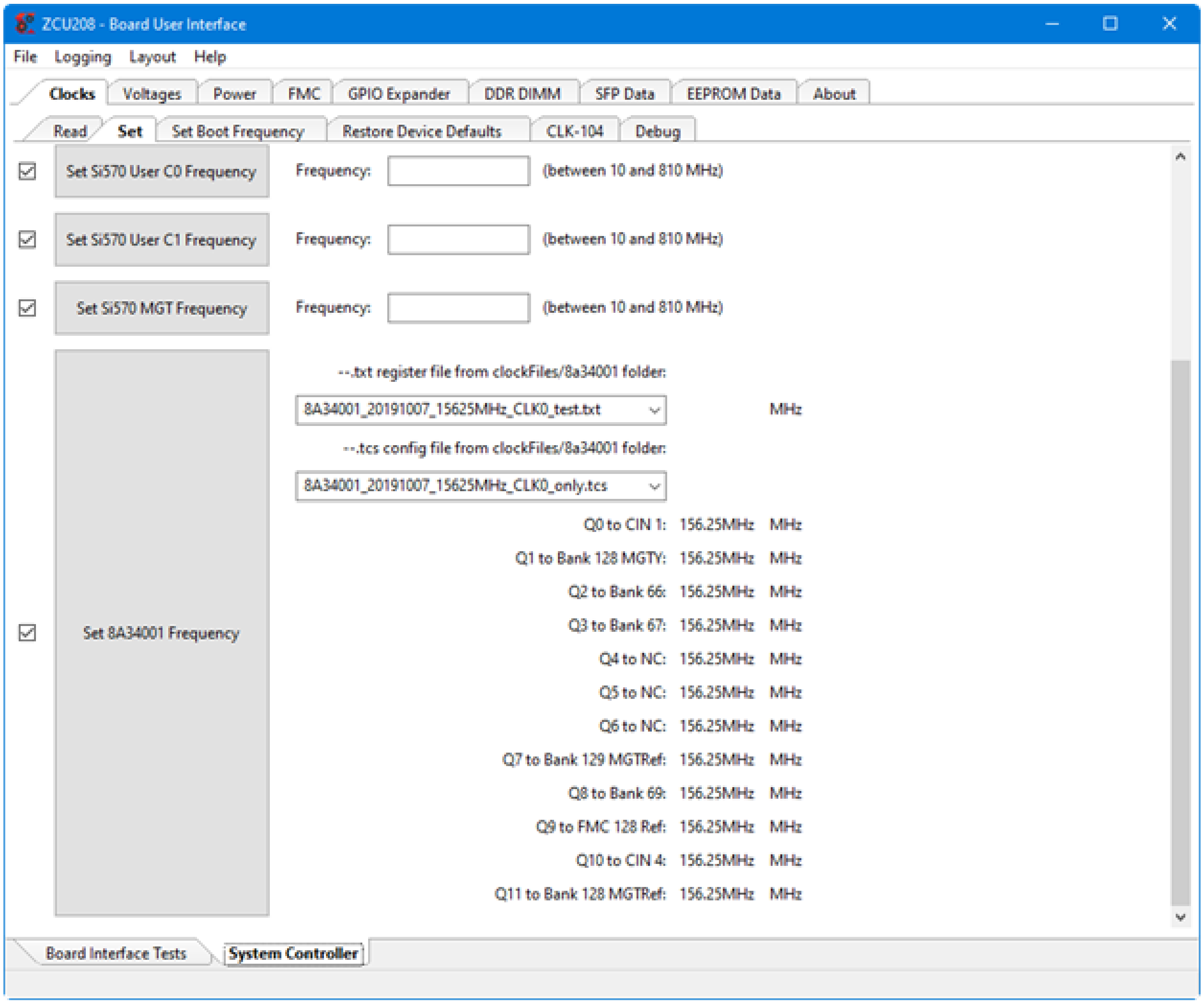


Transceiver Reference Clock Setup

The PLL must be programmed to generate 156.25 MHz. Files are provided with the tool. The following screenshot shows the setup to program the PLL 8A34001

IMPORTANT! *Configure the FPGA after the PLL. If the FPGA is configured before programing the PLL, a reconfiguration of the FPGA is required.*

Figure 10: **Transceiver Reference Clock Setup**



☑ Set Si570 User C0 Frequency — Frequency: (between 10 and 810 MHz)

☑ Set Si570 User C1 Frequency — Frequency: (between 10 and 810 MHz)

☑ Set Si570 MGT Frequency — Frequency: (between 10 and 810 MHz)

☑ Set 8A34001 Frequency

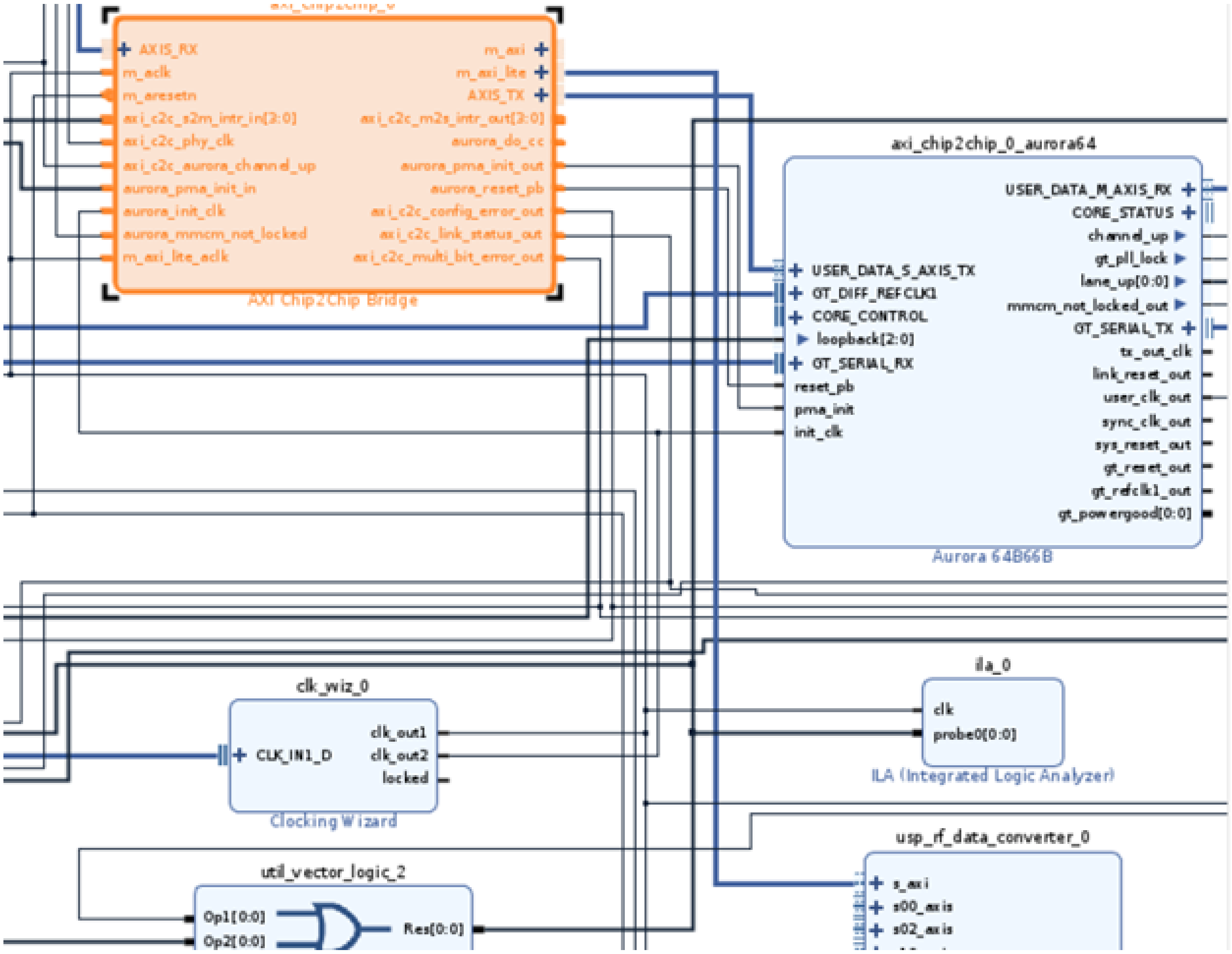
--.txt register file from clockFiles/8a34001 folder: 8A34001_20191007_15625MHz_CLK0_test.txt MHz

--.tcs config file from clockFiles/8a34001 folder: 8A34001_20191007_15625MHz_CLK0_only.tcs

Q0 to CIN 1:	156.25MHz	MHz
Q1 to Bank 128 MGTY:	156.25MHz	MHz
Q2 to Bank 66:	156.25MHz	MHz
Q3 to Bank 67:	156.25MHz	MHz
Q4 to NC:	156.25MHz	MHz
Q5 to NC:	156.25MHz	MHz
Q6 to NC:	156.25MHz	MHz
Q7 to Bank 129 MGTRef:	156.25MHz	MHz
Q8 to Bank 69:	156.25MHz	MHz
Q9 to FMC 128 Ref:	156.25MHz	MHz
Q10 to CIN 4:	156.25MHz	MHz
Q11 to Bank 128 MGTRef:	156.25MHz	MHz

The secondary design used on each of the ZCU208 boards is shown in the following figure.

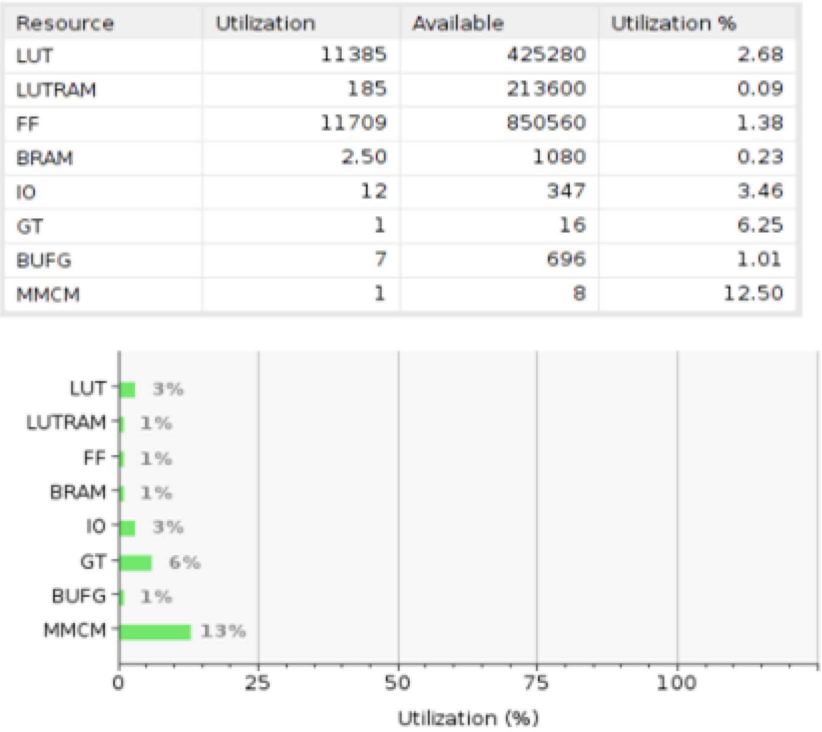
Figure 11: **Slave Chip2Chip Connection**



Utilization and Performance

Resources utilization for the secondary reference design is provided in the following figure.

Figure 12: **Utilization and Performance**



Resource	Utilization	Available	Utilization %
LUT	11385	425280	2.68
LUTRAM	185	213600	0.09
FF	11709	850560	1.38
BRAM	2.50	1080	0.23
IO	12	347	3.46
GT	1	16	6.25
BUFG	7	696	1.01
MMCM	1	8	12.50

FMC Debug Card

The FMC debug card routes the IRQ from the secondary RFDC to the primary controller.

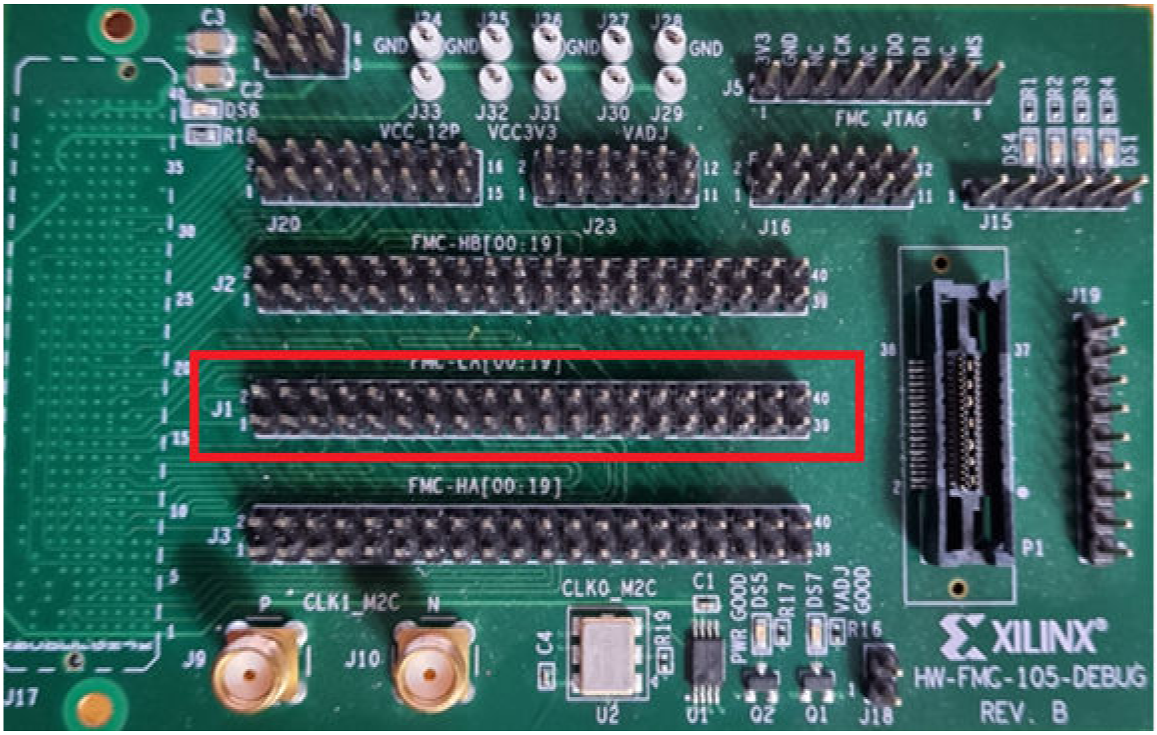
Primary Controller Board

The primary board receives two interrupt signals from the secondary boards. The interrupts are wired to the FMC debug card:

- IRQ[0] is wired to connector J1 pin 40
- IRQ[1] is wired to connector J1 pin 2

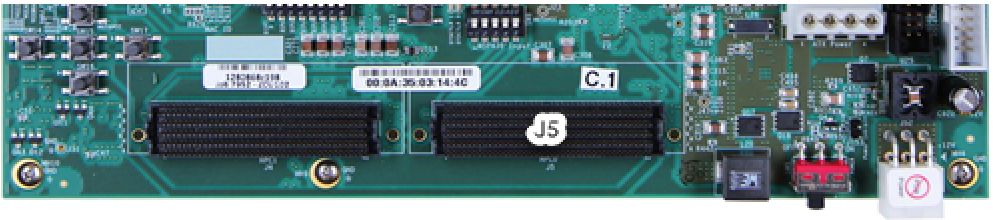
The FMC debug card identifying connector J1 is shown in the following figure.

Figure 13: **Connector J1**



The ZCU102 board has two FMC connectors, as shown in the following figure. The FMC debug card must be plugged into HPC0 connector J5.

Figure 14: **ZCU102 FMC Connectors**



The secondary RFDC interrupt is routed to the I/O signal on the FMC board. On the primary controller, the interrupts are routed to the GIC via shared peripheral interrupts (see the following table). In this reference design the RFDC interrupts are connected to the PL_PS group 0.

Table 6: **IRQ Interrupt Signals**

IRQ Name	RQ No. (GIC)	IDCICR	Bits	Required Type	GICPx_IRQ Bits (GIC Proxy)	Description
CSU	117	7	[11:10]	High level	GICP2 [21]	Configuration and security unit.
CSU_DMA	118	7	[13:12]	High level	GICP2 [22]	CSU DMA Controller.

Table 6: **IRQ Interrupt Signals** *(cont'd)*

IRQ Name	RQ No. (GIC)	IDCICR	Bits	Required Type	GICPx_IRQ Bits (GIC Proxy)	Description
eFuse	119	7	[15:14]	High level	GICP2 [23]	eFuse interrupt.
LPD_XMPU_XPPU	120	7	[17:16]	High level	GICP2 [24]	OCM XMPU and XPPU protection ints in LPD.
PL_PS_Group0	121-128	7-8	[31:18] [1:0]	Rising edge/High level	GICP2 [25-31] GICP3 [0]	PL to PS interrupt signals 0 - 7.

***Note*:** RFDC interrupts are connected to Vector 0 on the GIC. Linux ignores the 32 private interrupts and therefore the IRQ number is adjusted by 32:

- 121 - 32 = 89
- 122 - 32 = 90

Secondary Controller Board

The secondary board generates the IRQ interrupt signals to the primary board.

Vivado 2021.1 Hardware Design Setup

Unzip the hardware designs for the primary and secondary controllers. A tcl script is used to rebuild the Vivado block design. The Hwh from the secondary design is required for the primary petalinux device tree creation. The secondary Hwh files contain the RFDC configuration which are is required for the ‘parameter-list’ in the device tree.

tcl scripts are provided in the root folder to setup the projects:

- Primary: vivado -source primary_zcu102_x_.tcl
- Secondary: vivado -source secondary_zcu102_x_.tcl

Chapter 7

Linux Setup

The reference design includes the BSP configured for the zcu102, all receipes/patches, and software test applications.

RFDC 2021 Driver

The RFDC driver patch contains updates to support access to two instances of the RFDC.

RFDC Device Tree

The secondary hardware design contains the RFDC configuration details. These parameters must be extracted and used in the device tree for the primary controller. The primary design file supplied to the petalinux project has no information on the RFDC and does not have the device tree entry extracted for the RFDC when, using the normal flow.

The device tree must be provided as a dtsi file, as the BSP does contain the required entry for the RFDC. From the master design verify that the AXI memory address is connected to each slave RFDC. The following is the device tree template:

```
/include/ "system-conf.dtsi"
/ {
    amba_pl@0 {
        usp_rf_data_converter@a0000000 {
            compatible = "xlnx,usp-rf-data-converter-2.4";
            interrupt-parent = <&gic>
                               interrupts = <0 89 4>;
            param-list = paramlist1
            reg = <0x0 0xa0000000 0x0 0x40000>;
        };
        usp_rf_data_converter@a0040000 {
            compatible = "xlnx,usp-rf-data-converter-2.4";
            interrupt-parent = <&gic>
                               interrupts = <0 90 4>;
            param-list = paramlist2
            reg = <0x0 0xa0040000 0x0 0x40000>;
        };
    };
};
```

Secondary Design DT Parsing

The device tree generator .tcl has been patched in the BSP to add the secondary RFDC devices and insert them into the device tree.

Petalinux BSP

The reference design provides a BSP to help get started, using petalinux 2021.1. Extract the BSP using the following commands:

- petalinux-create -t project -s <path to BSP>.bsp -n <name>
- petalinux-build

BSP Files

- project-spec/meta-user/recipes-bsp/device-tree/files/system-user.dtsi
    - RFDC device tree
- project-spec/meta-user/recipes-bsp/rfdc
    - RFDC recipe, source and patch
- project-spec/meta-user/recipes-apps
    - rfdc-test
    - rfdc-irq

Booting the System

The tcl script ‘bootprebuilt.tcl’ downloads the pre-build images using JTAG. The Digilent JTAG-SMT2NC 210308AE700C must match the zcu102 board-specific serial number. Use the vivado tool's xsdb to connect to the board:

- xsdb% connect
- xsdb%jtag targets

You should see the following output:

```
1  Digilent JTAG-SMT2NC 210308AE700C
     2  xczu9 (idcode 24738093 irlen 12 fpga)
        3  bscan-switch (idcode 04900101 irlen 1 fpga)
           4  unknown (idcode 04900220 irlen 1 fpga)
     5  arm_dap (idcode 5ba00477 irlen 4)
  7  Xilinx ZCU208 FT4232H 05023a01041A
```

```
    8  unknown (idcode 047fb093 irlen 12)
    9  arm_dap (idcode 5ba00477 irlen 4)
11  Xilinx ZCU208 FT4232H 05023a01039A
   12  unknown (idcode 047fb093 irlen 12)
   13  arm_dap (idcode 5ba00477 irlen 4)
```

In the TCL file update 'jtag_cable_name =~' with the jtag target string in the output.

RFDC Software Applications

The reference design provides two linux test applications used to verify access to the RFDC:

- The RFDC R/W application is used to verify access to the RFDC, device id 0/1. It is a simple example of RFDC read/write verification. The following is the expected output:

  ```
  root@RemoteRFDC:~# rfdc-test
  Driver Version 10.100000
  Before - After
  Device 0 DAC 0 Fabric Rate: 14 -> 12
  Device 0 ADC 0 Fabric Rate: 8 -> 7
  Before - After
  Device 1 DAC 0 Fabric Rate: 14 -> 12
  Device 1 ADC 0 Fabric Rate: 8 -> 7
  Successfully ran Example
  root@RemoteRFDC:~#
  ```

- The RFDC IRQ test application is used to verify interrupt detection on the secondary RFDC devices. The application write access causes an internal RFDC error and the application interrupt handler detects the error. The following is the expected output:

  ```
  root@RemoteRFDC:~# rfdc-irq
  RFDC(0) Interrupts registered and enabled, now testing...
  INFO:RFDC(0) Interrupt Detected Correctly...
  RFDC(1) Interrupts registered and enabled, now testing...
  INFO:RFDC(1) Interrupt Detected Correctly...
  RFDC Interupt testing PASSED, both interrupts detected correctly...
  root@RemoteRFDC:~#
  ```

Appendix A

Additional Resources and Legal Notices

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see Xilinx Support.

Documentation Navigator and Design Hubs

Xilinx® Documentation Navigator (DocNav) provides access to Xilinx documents, videos, and support resources, which you can filter and search to find information. To open DocNav:

- From the Vivado® IDE, select **Help → Documentation and Tutorials**.
- On Windows, select **Start → All Programs → Xilinx Design Tools → DocNav**.
- At the Linux command prompt, enter `docnav`.

Xilinx Design Hubs provide links to documentation organized by design tasks and other topics, which you can use to learn key concepts and address frequently asked questions. To access the Design Hubs:

- In DocNav, click the **Design Hubs View** tab.
- On the Xilinx website, see the Design Hubs page.

Note**:** For more information on DocNav, see the Documentation Navigator page on the Xilinx website.

References

These documents provide supplemental material useful with this guide:

1. *Zynq UltraScale+ RFSoC Data Sheet: Overview* (DS889)
2. *Zynq UltraScale+ RFSoC Data Sheet: DC and AC Switching Characteristics* (DS926)
3. *ZCU102 Evaluation Board User Guide* (UG1182)
4. *ZCU208 Evaluation Board User Guide* (UG1410)
5. *AXI Chip2Chip LogiCORE IP Product Guide* (PG067)
6. *Aurora 64B/66B LogiCORE IP Product Guide* (PG074)
7. Clock Programming for xtp435
8. Clock Programming for xtp600

Revision History

The following table shows the revision history for this document.

Section	Revision Summary
03/16/2022 Version1.0	
Initial release.	N/A

Please Read: Important Legal Notices

The information disclosed to you hereunder (the "Materials") is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of Xilinx's limited warranty, please refer to Xilinx's Terms of Sale which can be viewed at https://

www.xilinx.com/legal.htm#tos; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx's Terms of Sale which can be viewed at https://www.xilinx.com/legal.htm#tos.

AUTOMOTIVE APPLICATIONS DISCLAIMER

AUTOMOTIVE PRODUCTS (IDENTIFIED AS "XA" IN THE PART NUMBER) ARE NOT WARRANTED FOR USE IN THE DEPLOYMENT OF AIRBAGS OR FOR USE IN APPLICATIONS THAT AFFECT CONTROL OF A VEHICLE ("SAFETY APPLICATION") UNLESS THERE IS A SAFETY CONCEPT OR REDUNDANCY FEATURE CONSISTENT WITH THE ISO 26262 AUTOMOTIVE SAFETY STANDARD ("SAFETY DESIGN"). CUSTOMER SHALL, PRIOR TO USING OR DISTRIBUTING ANY SYSTEMS THAT INCORPORATE PRODUCTS, THOROUGHLY TEST SUCH SYSTEMS FOR SAFETY PURPOSES. USE OF PRODUCTS IN A SAFETY APPLICATION WITHOUT A SAFETY DESIGN IS FULLY AT THE RISK OF CUSTOMER, SUBJECT ONLY TO APPLICABLE LAWS AND REGULATIONS GOVERNING LIMITATIONS ON PRODUCT LIABILITY.

Copyright

© Copyright 2022 Xilinx, Inc. Xilinx, the Xilinx logo, Alveo, Artix, Kintex, Kria, Spartan, Versal, Vitis, Virtex, Vivado, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. AMBA, AMBA Designer, Arm, ARM1176JZ-S, CoreSight, Cortex, PrimeCell, Mali, and MPCore are trademarks of Arm Limited in the EU and other countries. All other trademarks are the property of their respective owners.