Sail River Gen III (WS-VTP-00072)

Hardware Validation

Last updated: **2025-05-21**

Altera Confidential

Contents

List of Figures

List of Tables

1.0 Introduction

1.1 Objective

The objectives of this validation are:

- To provide the validation plan for the Sail River Gen III.
- To validate the design by injecting O-RAN packets from Keysight Oran Studio on the downlink side with the data digitally looped back on the uplink side, captured in Keysight Oran Studio, and plotted in MATLAB GUI.

1.2 Scope

The scope of this validation test plan document is to provide insight on the validation environment developed for the Sail River Gen III Module. The design is tested based on the data captured at various capture points in the MATLAB GUI.

Completion of validation will be based on the list of items below.

- Downlink is tested with O-RAN test packets sent from Keysight Oran Studio.
- Uplink is tested with digital loopback.
- Capture points are added at iFFT input/output, FFT input/output and DFE input/output and PRACH input/output, and can be captured and visualized in MATLAB.
- Uplink packets are received in Keysight Oran Studio.
- The received packets from Keysight Oran Studio are extracted and plotted in VSA.
- Normalized Mean Square Error (NMSE) and Error Vector Magnitude (EVM) are reported at each capturing stage.

1.3 Hardware Prerequisites

Table 1-1 Hardware Prerequisites

S.no	Device and Interface	Required Components		Part number	QTY	Comments /Dependencies
1	System components	GPS Antenna		101155-00	1	_
2		Grand Master		GM200	1	
3		Keysight Oran Studio		S5040A	1	_
4		CISCO Switch		Nexus 9000 C93810YC-FX3	1	-
5		HSSI Dev Kit - HiTek Board		_	1	-
6	Grand Master - Trimble GM-200	Ethernet (Eth2)	Ethernet Cable	RJ45	1	Eth2 - Management port for the configuration
7		SFP Module (Eth0)	10G SFP Module	FTLX8571D3BCV	1	Eth0 - Provides NTP/PTP connectivity to Ethernet Networks (connect with CISCO switch)

S.no	Device and Interface	Required Components		Part number	QTY	Comments /Dependencies
8		Serial Port (RS232)	RS232 to USB	HL340	1	Provides access to command line interface (CLI) (Needed for initial configuration in grand master)
9		SMA (Multi-GNSS Receiver)	Antenna cable (Coaxial Cable)	LMR-240-DB	1	Loss less Antenna Cable
10			Surge Protector	GXHM-TFF	1	Lightening Protector
11			TNC male to SMA male RG58 C/U coaxial cable	_	1	Connection between surge protector and GM
12			TNC Male connector	TC-240-TM-X	1	Connectors for antenna cable
13	CISCO Switch-	25G SFP28 Module		E25GSFP28SR	2	-
14		10G SFP Module		FTLX8571D3BCV	1	For PTP synchronization
15		Fiber optic cables		_	3	-
16	F Tile Dev Kit - HiTek Board	Ethernet Cable		RJ45	1	-
17		25G SFP Module		E25GSFP28SR	1	-
18		USB BLASTER II		PL-USB2-BLASTER	1	-
19		Si 5518 Timing Module MS-1		-	1	This timing module will be used for MS 1 and for Gen2/3
20	Renesas	RFCB3 Timing card (Renesas DPLL 38612) MS 1.5/2			1	RFCB3 Timing card will be replaced with Si5518 in MS1.5 OR 2
21		U. FL Connectors		931-1098-ND	2	-
22		Ethernet Cable		RJ45	1	-

S.no	Device and Interface	Required Components	Part number	QTY	Comments /Dependencies
23	Keysight Oran Studio	100G SR4 QSFP28 Transceiver	FTLC9558REPM	1	-
24		QSFP28 to SFP28 Multi mode Fiber breakout	-	1	-
25	RF Card	ADRV904x Koror EVM	ADRV9047	1	RF Evaluation Module
26	Cable	U.FL to SMA Cable	-	1	To connect between Si(J3) to RefA of AD9528(J901)

1.4 Software Prerequisites

Table 1-2 Software Prerequisites

S.no	Area/Type	Software	Version
1	Design SW	Quartus Prime Pro, including Qsys and DSP Builder	24.1
2	RTL simulation SW	Questa Full Edition-64	23.1
3	Validation (PC-Debug SW)	MATLAB	2023b (or later)
4	Validation (PC-Debug SW)	JRE	1.8.0.144-b01 (or later)
5	Validation (PC-Debug SW)	Wireshark with tshark	1.10.14
6	Validation (SW)	ClockBuilder pro	4.7 (or latest)
7	Validation (Server SW)	U5040B Open RAN studio	V3.1.11039
8	Validation (Server SW)	PathWave Vector Signal Analysis (89600 VSA)	2023_U2
9	Validation (Server SW)	PathWave Signal Generation	V5.0
10	Validation (PC-Terminal Emulator)	PuTTY/Tera Term	latest
11	OS	Yocto - Poky	Kirkstone
12	GCC	Open Souce/Yocto	gcc-11.3
13	ATF	Open Source/Yocto	2.11.0
14	UBoot	Open Source/Yocto	2024.04
15	Linux Kernel	Open Source/Yocto	6.1.38-lts
16	Validation (RF SW)	ACE , Analog Devices GUI	1.26.3240.1417 1.2023.42300(Board Plugin)

2.0 Test Setup

2.1 Trimble Grandmaster:

The Trimble Grandmaster User Guide is given below.
https://timing.trimble.com/wp-content/uploads/thunderbolt_PTP_GM200_user_guide.pdf

The Thunderbolt PTP GM has four communications ports:

- 1 Serial Port (RS232)
- 1 Management Port autosensing Ethernet (eth2) 10/100/1000 Base-T (RJ-45)
- 1 Traffic Port autosensing Ethernet (eth1) 10/100/1000 Base-T (RJ-45)
- 1 Traffic Port SFP(eth0) (Small Form-Factor Pluggable)

Either Serial port or Ethernet eth2 (RJ-45) is the dedicated management port to configure the GM.

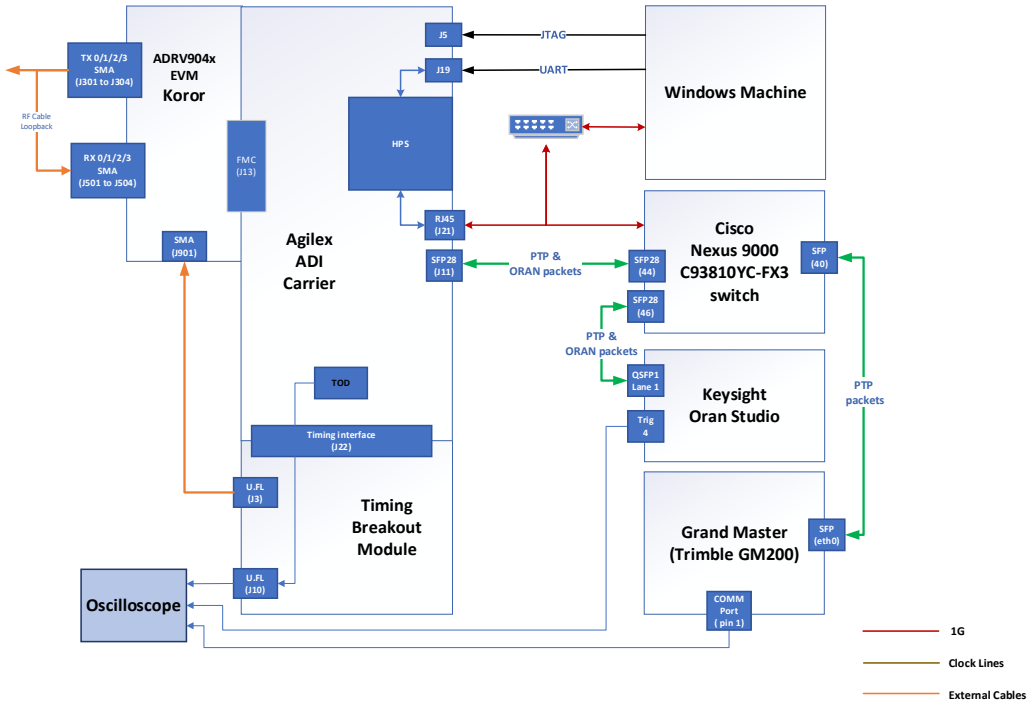
2.2 Hardware Setup

In Sail River, the mSGDMA data path and the 1588-related modules and drivers required to provide HSSI Ethernet and 1588 HW support to the HPS software stack are available. The available hardware allows for the S-Plane to be received over the SFP28 25GbE link established between the O-DU and the O-RU, together with the C- and U-plane.

In this case, the Si5518 Timing module will be connected through timing interface(J22) of the ADI carrier Hitek card. It receives a clock input (PMA RX recovery clock for SyncE support) from the FPGA HSSI, uses this clock to lock its SyncE and 1588 DPLLs to provide the HSSI reference clock and the 1588-disciplined clock back to the Hitek's Agilex FPGA. The HPS SPI bus can interface the Si5518 programming interface via two board's SPI connections.

The current version of Gen3 supports Si5518, which is the Clock Master. AD9528 Clock Generator provides the Clock and Sysref to both Koror and FPGA but AD9528 is synchronized with Si5518 by providing the external reference of 122.88 MHz to J901 from J3(Out8 of Si5518). Other clocks were routed to HiTek Agilex eSOM from Si5518 such as ToD clock, CDR div66 recovery clock, HSSI System PLL reference clock.

Figure 2-1 Hardware Setup



The network connectivity resembles an LLS-C3 Topology, from a synchronization point of view. In that case, Trimble GM200 becomes the primary timing source (T-GM) and the O-RU (eSOM card) and O-DU (Keysight Oran Studio) is the ordinary clock connected through Cisco switch which acts as boundary clock.

The Trimble GM200 grandmaster can output a 1PPS for test measurements and the 1PPS from the FPGA ToD can be routed to an oscilloscope through a jumper connector on the eSOM card and 1PPS from Keysight ORAN studio can be routed to an oscilloscope through a SMA connector.

- The HPS terminal is accessed via eSOM UART.
- The Si5518 chip is programmed over SPI to produce the required output clocks, 156.25MHz, etc.
- AD9528 provides Clock and Sysref to both FPGA and Koror. But synchronized with Si5518 with a reference input of 122.88 MHz.
- The Keysight Oran Studio transfers the C-plane, U-plane and S-plane packets from to Agilex eSOM through 25GbE (SFP28) interface.
- The M-plane packets are carried through 1GbE ethernet interface from Matlab to Agilex eSOM for test purposes.

3.0 HW Validation Test Cases

3.1 Sanity Test Cases

Table 3-1 summarizes different test cases for board bring up and initial sanity checks.

S. No	Tests	Description
1	Initial board bring up with HPS booting	Booting HSSI Agilex eSOM (HiTek board) with all the FPGA and SW binaries derived from GHRD and GSRD
2	HSSI 25GbE Optical Loopback test in HiTek Board	Loopback testing will be performed in the HiTek Board, in SFP28 link. Packets are injected into MAC TX and received in MAC Rx with external Optical loopback.
3	25GbE Linkup test between ODU (Keysight Oran Studio) and ORU (HiTek board)	Ethernet 25G linkup test should be carried out between, the Agilex eSOM SFP28 (J9) and the Keysight Oran studio (QSFP1 port 1) to ensure the establishment of connection between them.
4	ORAN loop-back testing	ORAN traffic is sent from the Keysight Oran studio and looped back from DL to UL at ORAN IPs output. By default, looping back at ORAN would validate the eCPRI, ORAN, and the Ethernet connectivity IPs.
5	1G Ethernet in HPS – Ping Test	Ping test will be carried out between MATLAB server to HiTek Board for testing 1G connections
6	MATLAB Communication	To establish the TCP/IP socket connection between the MATLAB and SoC application in the HiTek Board for configuration and capturing the data

Table 3-1 Board bring-up Test

3.2 Synchronization Test Cases

Table 3-2 Synchronization Test Cases

S. No	Tests	Description
1	Si5518/Renesas 38612 Chip Configuration from HPS	HPS SPI access test and clock configuration for HSSI reference clock and PMA recovery clock.
2	Synchronization Test	The O-RU synchronisation on the Agilex eSOM card uses linux-ptp. The 1 PPS generated from ToD IP is observed in an Oscilloscope together with the 1 PPS reference from the Trimble GM200 (Grandmaster) and 1 PPS from Keysight ORAN studio and measure the time offset and assure synchronization accuracy

3.3 Datapath and Module level testing

Table 3-3 Datapath Testing

S. No	Tests	Description
1	All capture points at DL and UL line up	Every module is validated by capturing 1 symbol and comparing against the expected vector. NMSE value of the test decides whether the test is passed or failed. Other key metrics like EVM, power and PAPR will also be measured in Matlab as part of signal quality measurements.

4.0 MATLAB GUI

4.1 Register Configuration page

To support running Sail River O-RU standalone or without Matlab support, all initial register configurations are done by the firmware application and the configured values will be reflected in the MATLAB GUI. It allows users to override the default configuration anytime. Register level access to all components are populated in GUI for debugging.

Register configuration override is done through MATLAB GUI to eSOM through 1G TCP/IP communication. M-plane configuration in the MATLAB GUI has Two tabs.

- **eCPRI ORAN SS: -** The eCPRI ORAN SS has the configuration for Test case selection e-CPRI, ORAN, EAXC ID as shown in Figure 4-1
- **DXC SS**: - The DXC SS tab has the Configuration for DXC, DUC, CA interp config, dec dly compr config and DDC config registers as shown in Figure 4-2
- **LOW PHY SS: -** The Low phy ss tab has the Configuration for PRACH, IFFT,FFT and Low phy configs as shown in Figure 4-3

The configuration should follow the following sequence:

1. eCPRI, ORAN, FH Compr & EAXC ID registers.
2. DXC,DUC, CA interp, decimator delay comp and DDC registers.
3. PRACH ,IFFT,FFT Low phy registers.

The GUI has ***Default and Configure*** button. When the GUI opens all parameters will be in default state. The Reset state also contains all default parameters and when the ***Default*** is clicked all parameters will be reverted to the default working parameters.

The ***Configure*** button sends all parameters from MATLAB to Agilex eSOM in TCP/IP packets. Each configuration is sent as an individual packets.

Figure 4-1 eCPRI ORAN tab

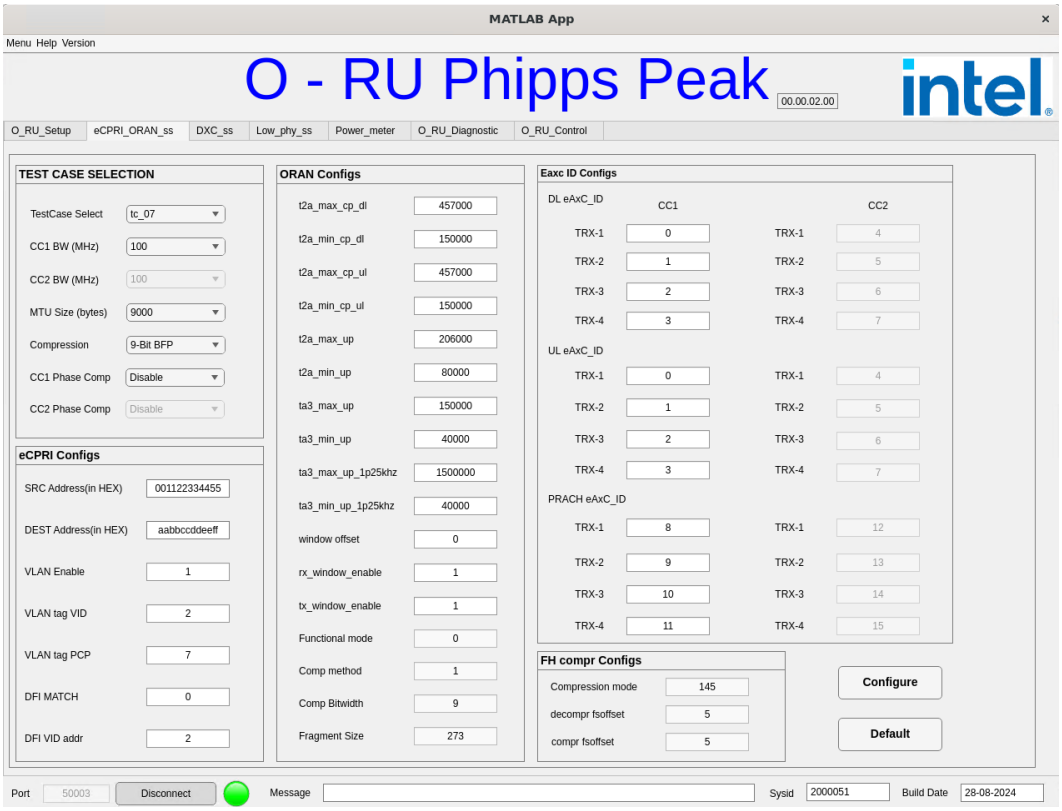


Figure 4-2 DXC tab

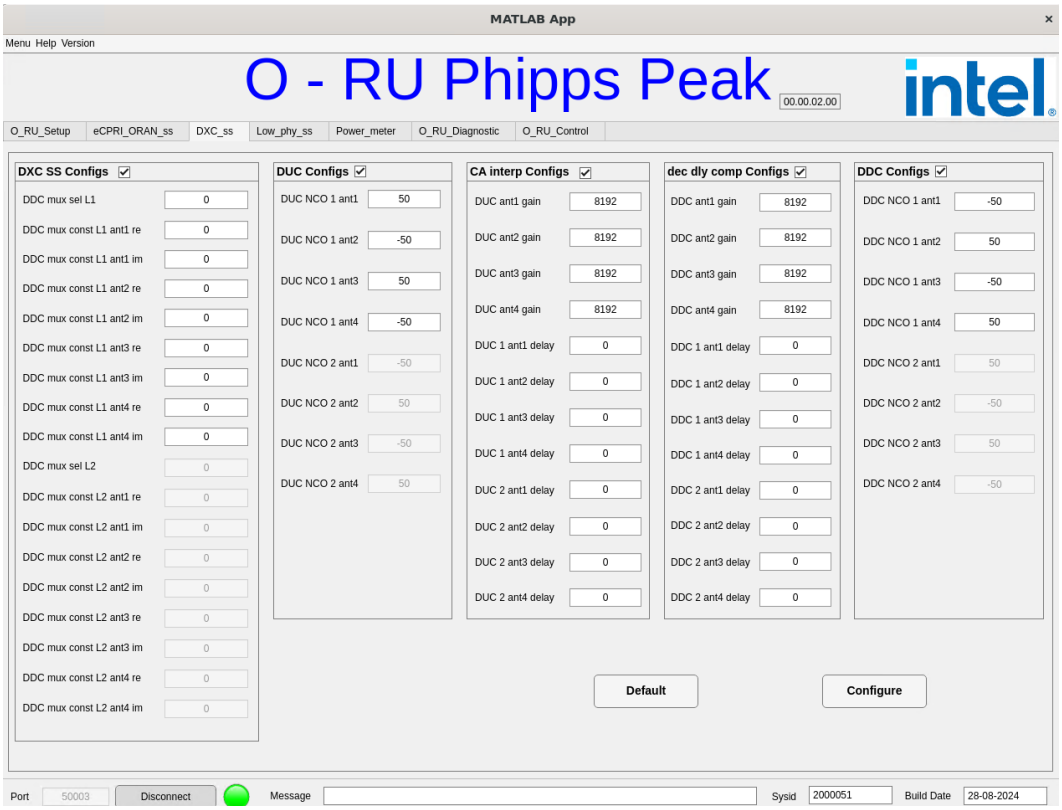
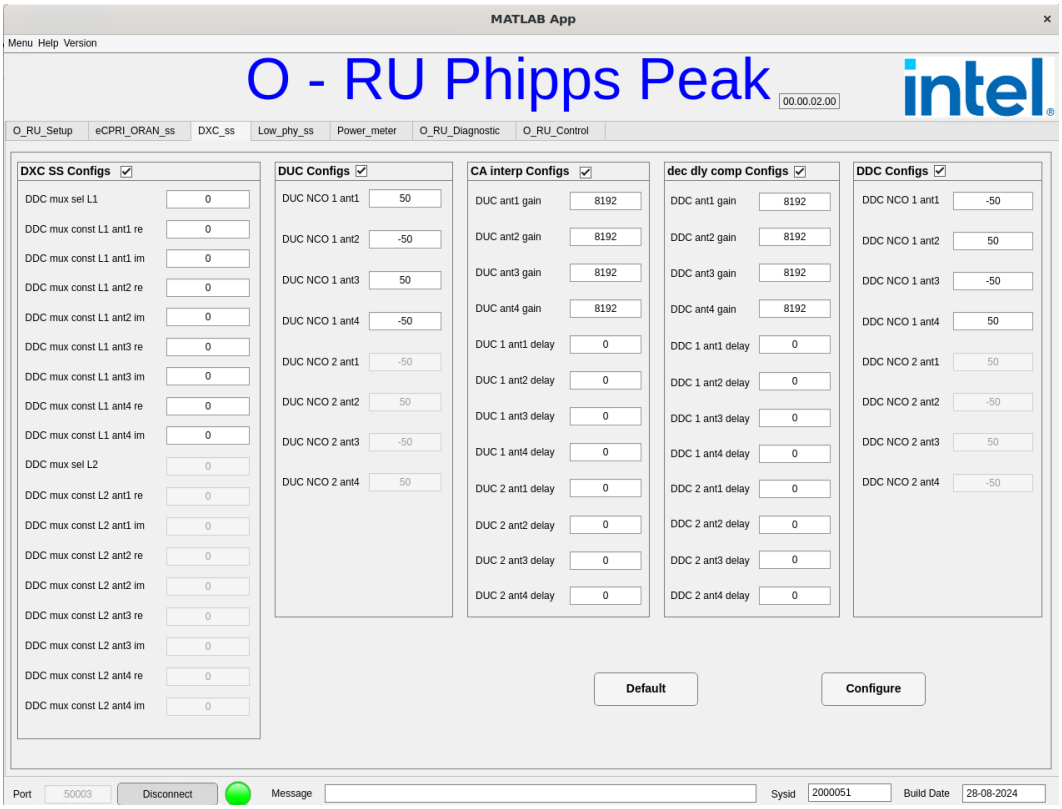
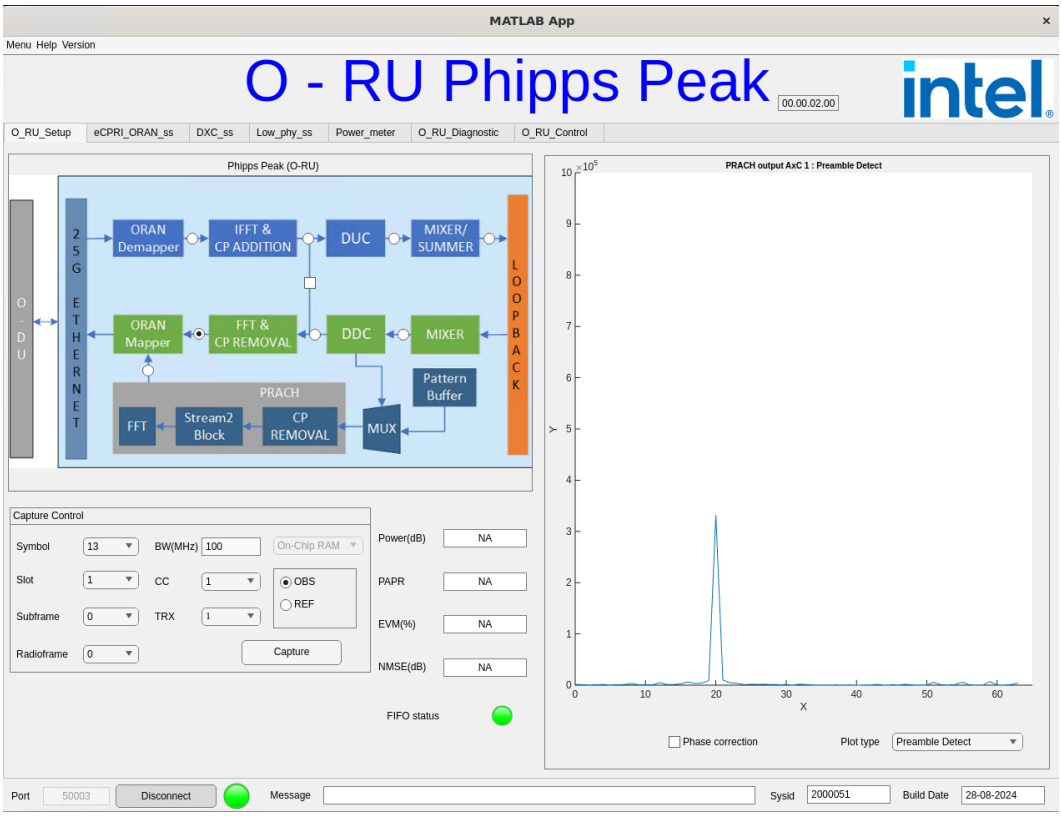


Figure 4-3 Low phy ss tab



O_RU_Setup tab **Error! Reference source not found.** shows the MATLAB user i nterface set up page to probe and capture at different stages of the transmit and receive chain. The tab is designed for functional testing and validations. Below we describe different features of the setup page.

Figure 4-4 MATLAB GUI setup tab



- ***Bandwidth Configuration:***

Band width can be configured per CC at "O_RU_Setup" Tab in GUI as shown in Figure 4-4

- ***Capture control***

The Capture panel has parameters for capturing one subframe based on CC and TRx. If REF is enabled, it plots the reference data from .mat file. If OBS is enabled, it gets the data from Agilex eSOM and plots the data.

- ***Plot***

The captured data is plotted in the figure at right side. Both Time-domain and Frequency -domain data can be plotted by choosing the plot type option. Plotting entire subframe and symbol by symbols options are available for key parameter analysis (ex EVM)

- ***Pattern Control***

The Pattern Control has the tools to inject data to pattern buffer for PRACH processing. PRACH test data for arbitrary Formats are generated in Matlab using 5G toolbox and uploaded in the pattern buffer for testing.

- ***Connection Panel***

The Connection panel has the port number text box, connect button and 1Gbe indicator for linkup.

- ***Port:***

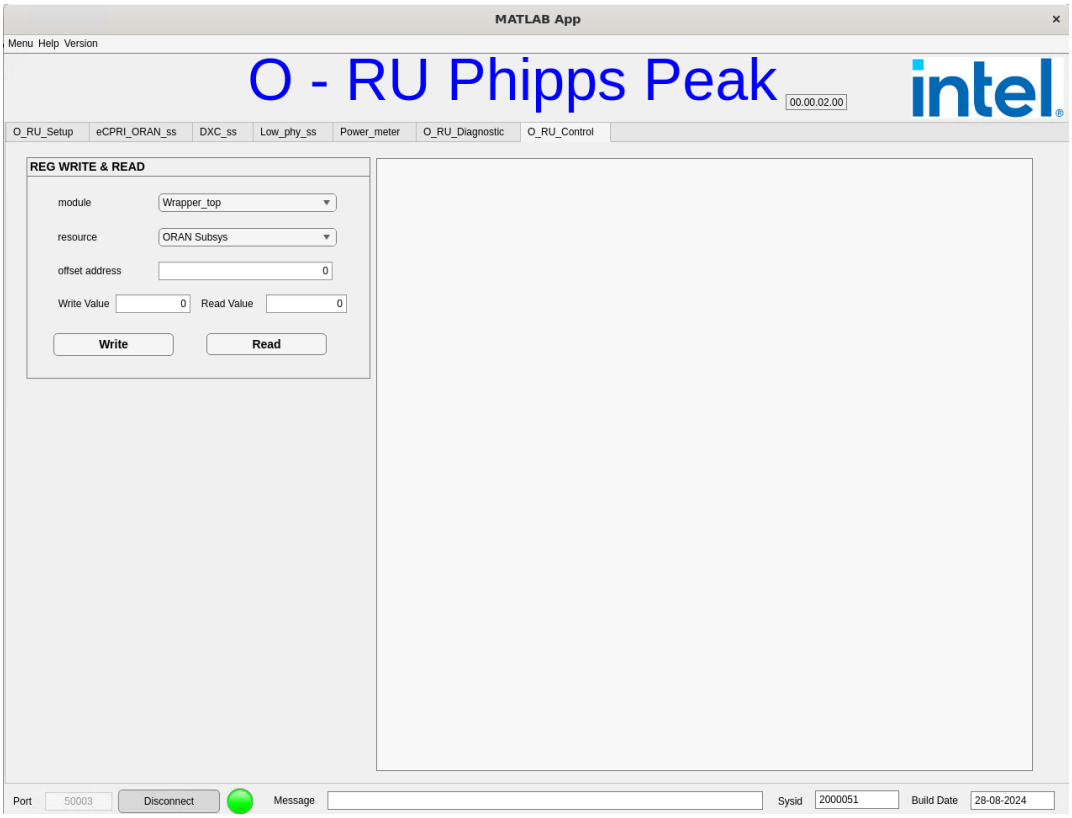
  The port number should be given the same as the number given in SoC application running in Agilex eSOM.

- ***Connect to SoC push button:***

  Establishes the connection between Matlab and SoC application in HPS.

The control tab **Error! Reference source not found.** is common control used for r egister read and write of sysid and wrapper_top modules. In this tab we need to select the module and resource and then enter the offset address to read or write. Enter the Value to write in that offset address in Write Value text box and click **"Write"** button to write the register. To read the register Click **"Read"** button the read value will be displayed in Read Value text box.

Figure 4-5 MATLAB GUI control tab

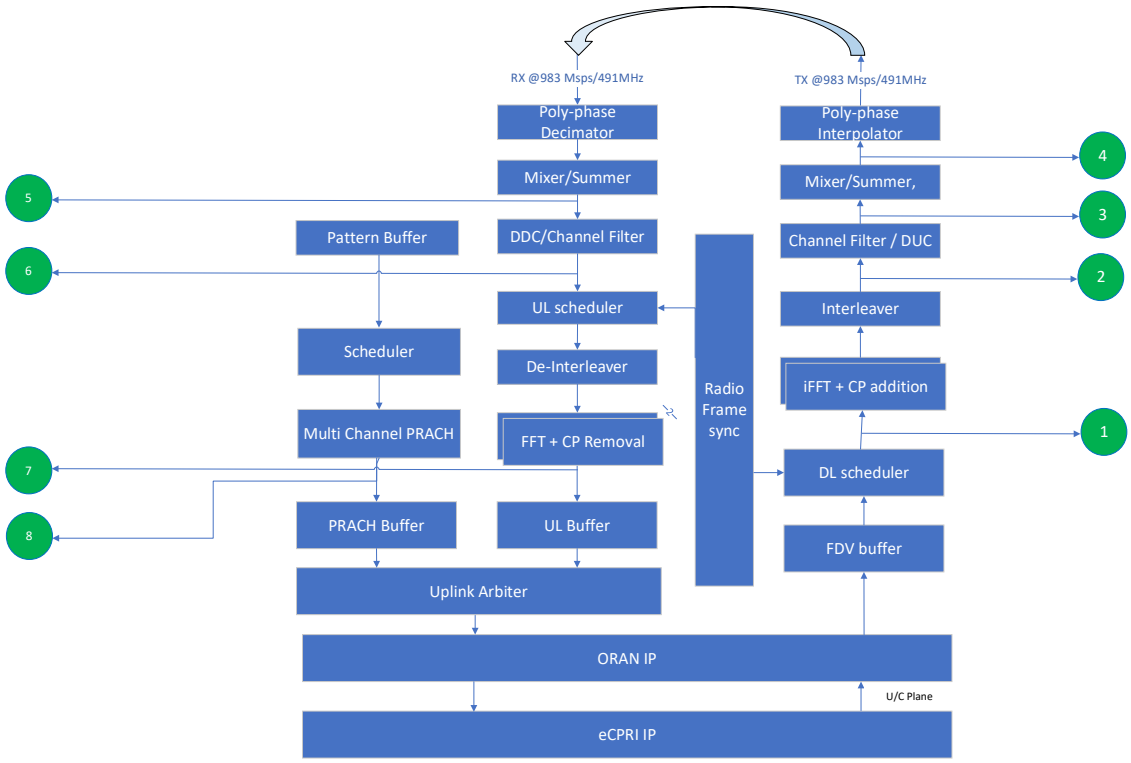


5.0 System-Level Test Scenarios based on data capturing in Matlab GUI

Different test scenarios are defined in this chapter to verify Sail River system behavior. These tests are done through MATLAB GUI capturing of test waveform in different stages of DL and UL. Data capturing is done in Agilex SoC using Sail River MATLAB GUI as described above. In the DL processing, data is captured at the input and output of IFFT+CP addition, as well as DFE input and output. In the UL processing, data is captured at input and output of DFE and FFT+CP removal, and eventually, the uplink packets from the Agilex SoC are recorded in Keysight Oran studio. All data captured are extracted and plotted in VSA.

Capture interface for Sail River with capture size of one symbol using On-chip RAM. The capture granularity is symbol, and the capture size shall be configured using GUI. Capture points 1 to 8 (as shown in **Error! Reference source not found.** 5-1) are c aptured at the DSP Clock domain and capture timing is calculated with respect to radio frame sync timer.

Figure 5-1 Capture points

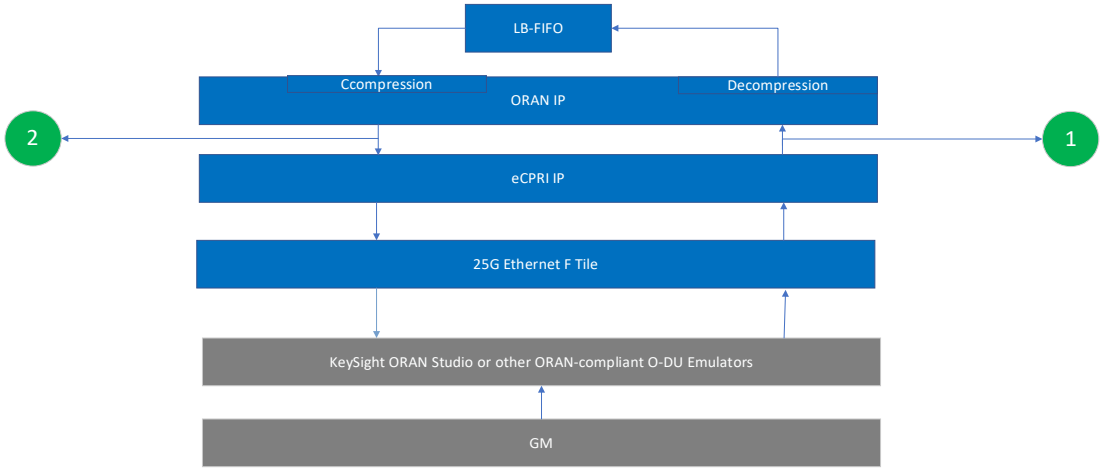


5.1 Register Read/Write Configuration

Registers for different modules, e.g., eCPRI, Channel filter, NCOs, etc, will be programmed with testing values and the results will be studied and verified.

5.2 ORAN Loopback Testing without Coupling

Figure 5-2 ORAN Loopback Capture Points



Abstract

The objective of the testing is to validate the compression, Fragmentation based on the MTU Size, Compression with fragmentation.

Test Setup:

Figure 5-3 ORAN-IP Configuration

Parameter	Value	Description
Maximum Ethernet frame SIZE	1500/9000	Compile time configuration, set MTU to 9000 and MTU is limited with programmable feature
Enable U-plane fragmentation	On	

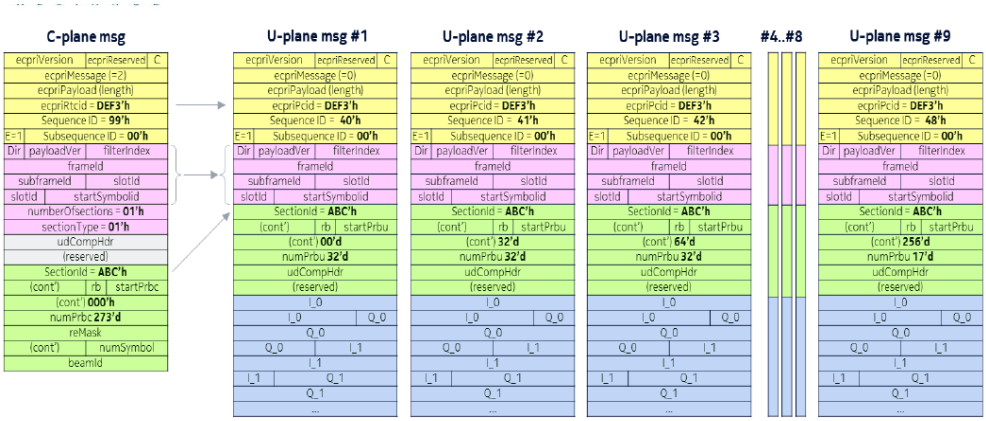
Parameter	Value	Description
Programmable U-Plane Fragmentation Size	On	MTU size validated for 1500 and 9000
Enable c-plane fragmentation	On	
Programmable c-Plane Fragmentation Size	On	MTU size validated for 1500 and 9000

Test vector is generated in PATHWAVE signal generator comprises of compression enable/disable, Single section and multiple section ORAN packets. Configure the Sail River O-RU ORAN IP MTU size to 1500, Loopback mode enabled. Schedule the ORAN packet from the Keysight ORAN simulator. Capture the loopback data in ORAN STUDIO and check for de-modulation and inspect the header for the correctness. Initially the check is done manually and later it can be automated.

ORAN-IP Configuration

Features covered in this testing:

1. Sequence ID check
    a. SeqID field E should be always set to 1 and Subsequence ID to 0.
    b. Sequence ID starts from 0 for the first fragment and counts for each fragment.
    c. Section ID should be maintained in the fragmented packet. It should be independent of C-Plane and U-Plane sequence id.



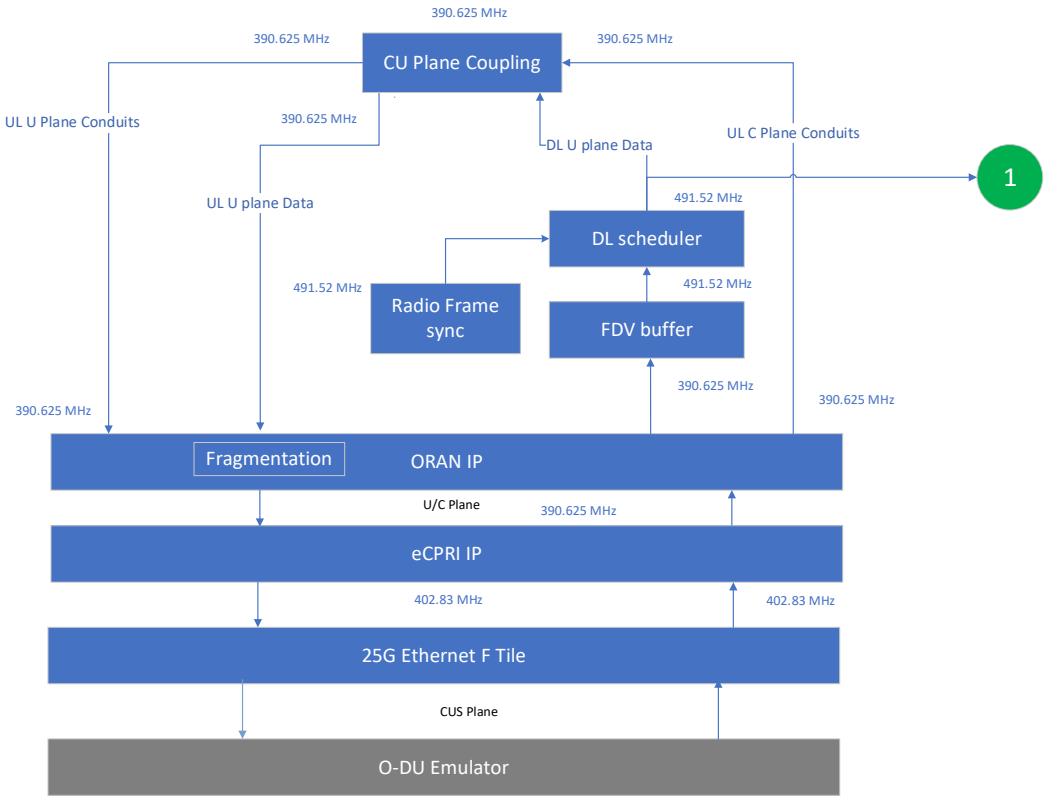
2. U-Plane fragmentation
3. CU-Plane fragmentation with compression
4. Fragmentation with multiple section
5. 100MHz
6. 60MHz

ORAN-IP Limitation

1. MTU size configured in PRB's and not BYTES.
2. The current version of the ORAN-IP splits each section into one Ethernet packet, Fragmentation with multiple section is tested with limitation.

5.3 CU-Plane Coupling

Figure 5-4 Capture Points to test CU- Plane Coupling



Abstract:

The objective of this testing is to validate the CU-Plane coupling based on section ID along with compression, Fragmentation based on the MTU Size, Compression with

fragmentation. Capture points 1 to 4 shown in **Error! Reference source not found.** h elps to identify coupling and fragmentation issues.

Testing Strategy

Keysight ORAN studio supports waveform generation for CU-Plane coupling based on section ID. It generates the following ORAN packets.

1. DL C-Plane Packet (DL PCAP)
2. DL U-Plane Packet (DL PCAP)
3. UL C-Plane Packet (DL PCAP)
4. UL U-Plane Packet (UL PCAP, separate file)

CU-Plane coupling can be tested in two different modes Loopback Mode and Uplink data injection mode. For Milestone-1 release, the plan is to validate CU-Plane coupling feature in loopback mode and Uplink data injection mode is an option.

1. Loopback Mode

   Schedule CU-Plane Packets from Keysight ORAN studio. CU-Plane coupling block parses the Uplink C-Plane and stores metadata in the C-Plane FDV buffer. During the uplink coupling block, frame the loopback data based on the coupling information available in the C-Plane FDV buffer. The framed data is expected to match with the generated downlink PCAP (Except ORAN packet direction field).

5.3.1 Test Case 1: Single Section

Test case 1 validates the single section functionality. The Cplane description of test case 1 is given in Table 5.

In this test case, the following settings are used.

1. Bandwidth is 100Mhz. [using dl_ul_bw_config register]
2. CC1 and CC2 are enabled. [using dl_ul_bw_config register]
3. Long PRACH Format 0. SF 1,3,5,7,9

Table 5-1 Test Case 1 Cplane Description

PKT	Slot	start Symbol	End symbol	Num Symbol	start PRB	End PRB	Num PRB
1	0	0	13	14	0	272	273
2	1	0	13	14	0	272	273
3	2	0	13	14	0	272	273
4	3	0	13	14	0	272	273

5.3.2 Test Case 2: Single Section, 1 Cplane packet per Symbol

This test case contains 1 cplane packet for every symbol. There are 14 cplane packets for 14 symbols in a slot. Same Cplane description tested for 4 slots. The Cplane description of test case 2 is given in Table 8.

In this test case, the following settings are used.

4. Bandwidth is 100Mhz. [using dl_ul_bw_config register]
5. CC1 and CC2 are enabled. [using dl_ul_bw_config register]
6. Long PRACH Format 0. SF 1,3,5,7,9

Table 5-2 Test Case 2 Cplane Description

Packet	Slot	start Symbol	End symbol	Num Symbol	start PRB	End PRB	Num PRB
1	0	0	0	1	0	272	273
2	0	1	1	1	0	272	273
3	0	2	2	1	0	272	273
4	0	3	3	1	0	272	273
5	0	4	4	1	0	272	273
6	0	5	5	1	0	272	273
7	0	6	6	1	0	272	273
8	0	7	7	1	0	272	273
9	0	8	8	1	0	272	273
10	0	9	9	1	0	272	273
11	0	10	10	1	0	272	273
12	0	11	11	1	0	272	273
13	0	12	12	1	0	272	273
14	0	13	13	1	0	272	273

5.3.3 Test Case 3: Multiple Sections

Test case 3 contains multi-section Cplane packets. Each section has 13 PRBs and there are 21 sections. Same Cplane description tested for 4 slots. The Cplane description of test case 3 is given in Table 9.

In this test case, the following settings are used.

7. Bandwidth is 100Mhz. [using dl_ul_bw_config register]
8. CC1 and CC2 are enabled. [using dl_ul_bw_config register]
9. Long PRACH Format 0. SF 1,3,5,7,9

Table 5-3 Test Case 3 Cplane Description

Packet	Slot	start Symbol	End symbol	Num Symbol	start PRB	End PRB	Num PRB
1	0	0	13	14	0	12	13
2	0	0	13	14	13	25	13
3	0	0	13	14	26	38	13
4	0	0	13	14	39	51	13
5	0	0	13	14	52	64	13
6	0	0	13	14	65	77	13
7	0	0	13	14	78	90	13
8	0	0	13	14	91	103	13
9	0	0	13	14	104	116	13
10	0	0	13	14	117	129	13
11	0	0	13	14	130	142	13
12	0	0	13	14	143	155	13
13	0	0	13	14	156	168	13
14	0	0	13	14	169	181	13
15	0	0	13	14	182	194	13
16	0	0	13	14	195	207	13
17	1	0	13	14	208	220	13
18	1	0	13	14	221	233	13
19	1	0	13	14	234	246	13
20	1	0	13	14	247	259	13
21	1	0	13	14	260	272	13

5.3.4 Test Case 4: TDD Mode

Test Case 4 validates the TDD special slot pattern. In a Radio Frame there are 10 Subframes. Each Subframe contains 2 slots. There are 20 Slots in 1 Radio Frame.

The TDD Pattern is as follows: DDDSUUDDDDDDDSUUDDDD.

The uplink packets are present in SF 1,2,6,7. Here Subframe 1 slot 1 and Subframe 6 Slot 1 are special slots. They contain 6 Symbols of downlink, 4 Symbols of guard band and 4 symbols of uplink data.

The following settings are used in this test case.

10. Bandwidth is 100Mhz. [using dl_ul_bw_config register]
11. CC1 enabled, CC2 disabled. [using dl_ul_bw_config register]
12. Long PRACH Format 0, Config Index 19. SF 2,7.

Table 5-4 Test Case 4 Cplane Description

Packet	Slot	start Symbol	End symbol	Num Symbol	start PRB	End PRB	Num PRB	AXCs Blanked	AXCs Not Blanked
	0,1,2	0	13	14	0	272	273	0,1,2,3,4,5,6,7	-
1	3	10	13	4	0	272	273	1,3,5,6,7	2,4,6
2	4	0	13	14	0	272	273	1,3,5,7	0,2,4,6
3	5	0	13	14	0	272	273	1,3,5,7	0,2,4,6
	6,7,8,9,10,11,12	0	13	14	0	272	273	0,1,2,3,4,5,6,7	-
4	13	10	13	4	0	272	273	1,3,5,6,7	2,4,6
5	14	0	13	14	0	272	273	1,3,5,7	0,2,4,6
6	15	0	13	14	0	272	273	-	0,2,4,6

5.3.5 Test Case 5: Blanking Modes

There are three different types of blanking, namely eAxC Blanking, Symbol Blanking, PRB Blanking. The blanking scenarios can occur as standalone or in combination with each other.

5.3.5.1 Standalone eAxC /Blanking

Test Case 5.1 validates the eAxC blanking feature. The DU can request only 1-7 eAxCs out of the 8 Uplane eAxCs. The CU Plane coupling IP must forward only those eAxCs requested by the DU. The Cplane description is given in Table 12.

The following settings are used in this test case.

13. Bandwidth is 100Mhz. [using dl_ul_bw_config register]
14. CC1 enabled, CC2 enabled. [using dl_ul_bw_config register]
15. Long PRACH Format 0, Config Index 27. SF 1,3,5,7,9.

Packet	Slot	Start Symbol	End symbol	Num Symbol	start PRB	End PRB	Num PRB	AXCs Blanked	AXCs Not Blanked
1	0	0	13	14	0	129	130	1,3,5,6	0,2,4,7

Packet	Slot	Start Symbol	End symbol	Num Symbol	start PRB	End PRB	Num PRB	AXCs Blanked	AXCs Not Blanked
2	0	0	13	14	130	272	143	1,3,5,6	0,2,4,7
3	1	0	13	14	0	129	130	0,2,4,7	1,3,5,6
4	1	0	13	14	130	272	143	0,2,4,7	1,3,5,6
5	2	0	13	14	0	90	91	0,3,5,7	1,2,4,6
6	2	0	13	14	91	181	91	0,3,5,7	1,2,4,6
7	2	0	13	14	182	272	91	0,3,5,7	1,2,4,6
8	3	0	13	14	0	272	273	-	0,1,2,3,4,5,6,7

Table 5-5 Test Case 5.1 Cplane Description

5.3.5.2 Standalone Symbol Blanking

Test Case 5.2 validates the Symbol blanking feature. The DU can request any of the 0-13 symbols. The CU Plane coupling IP must forward only those symbols requested by the DU. The Cplane description is given in Table 13.

The following settings are used in this test case.

16. Bandwidth is 100Mhz. [using dl_ul_bw_config register]
17. CC1 enabled, CC2 enabled. [using dl_ul_bw_config register]
18. Long PRACH Format 0, Config Index 27. SF 1,3,5,7,9

Table 5-6 Test Case 5.2 Cplane Description

Packet	Slot	Start Symbol	End symbol	Num Symbol	start PRB	End PRB	Num PRB
1	0	0	2	3	0	99	100
2	0	0	2	3	100	272	173
	0	3	13	11	0	272	273
	1	0	2	3	0	272	273
3	1	3	6	4	0	90	91
4	1	3	6	4	91	181	91
5	1	3	6	4	182	272	91
	1	7	7	1	0	272	273
6	1	8	13	6	0	19	20
7	1	8	13	6	20	39	20
8	1	8	13	6	40	272	233
	2	0	3	4	0	272	273
9	2	10	13	4	0	129	130

Packet	Slot	Start Symbol	End symbol	Num Symbol	start PRB	End PRB	Num PRB
10	2	10	13	4	130	272	143
	2	8	13	6	0	272	273
11	3	0	13	14	0	272	273

5.3.5.3 eAxC and Symbol Blanking

Test Case 5.3 validates both eAxC and Symbol blanking simultaneously. The DU can request any of the 0-13 symbols and any eAxC from 0-7. The CU Plane coupling IP must forward only those symbols and eAxCs requested by the DU. The Cplane description is given in Table 13.

The following settings are used in this test case.

19. Bandwidth is 100Mhz. [using dl_ul_bw_config register]
20. CC1 enabled, CC2 enabled. [using dl_ul_bw_config register]
21. Long PRACH Format 0, Config Index 27. SF 1,3,5,7,9

Table 5-7 Test Case 5.3 Cplane Description

Packet	Slot	Start Symbol	End symbol	Num Symbol	Start PRB	End PRB	Num PRB
1	0	0	2	3	0	49	50
	0	0	2	3	50	99	50
2	0	0	2	3	100	272	173
	0	3	13	11	0	272	273
	0	0	7	8	0	272	273
3	1	8	13	6	10	29	20
	1	8	13	6	30	39	10
4	1	8	13	6	40	59	20
5	1	8	13	6	60	272	213
	2	0	3	4	0	272	273
6	2	4	7	4	0	129	130
	2	4	7	4	130	272	143
	2	8	13	6	0	272	273
	3	0	13	14	0	49	50
7	3	0	13	14	50	149	100
	3	0	13	14	150	272	123

5.4 Windowing

Transmission and Reception window can be derived as

Table 5-8 Transmission and Reception Windowing

Downlink		
Window	**T1a_Max_Up (us)**	**T1a_Min_Up (us)**
U-Plane	206	135
C-Plane	457	205
Uplink		
Window	**Ta4_min (us)**	**Ta4_max (us)**
U-Plane	40	205
C-Plane	457	205
	Prach	
Window	**Ta4_min (us)**	**Ta4_max (us)**
U-Plane	40	1055
C-Plane	457	205

	Earliest Reception at O-DU	Latest Reception at O-DU
U-Plane	Ta4min ≤ Ta3min + T34min	Ta4max ≥ Ta3max + T34max
C-Plane	T1a_min_cp_ul ≥ T2a_min_cp_ul + T12_max	T1a_max_cp_ul ≤ T2a_max_cp_ul + T12_min

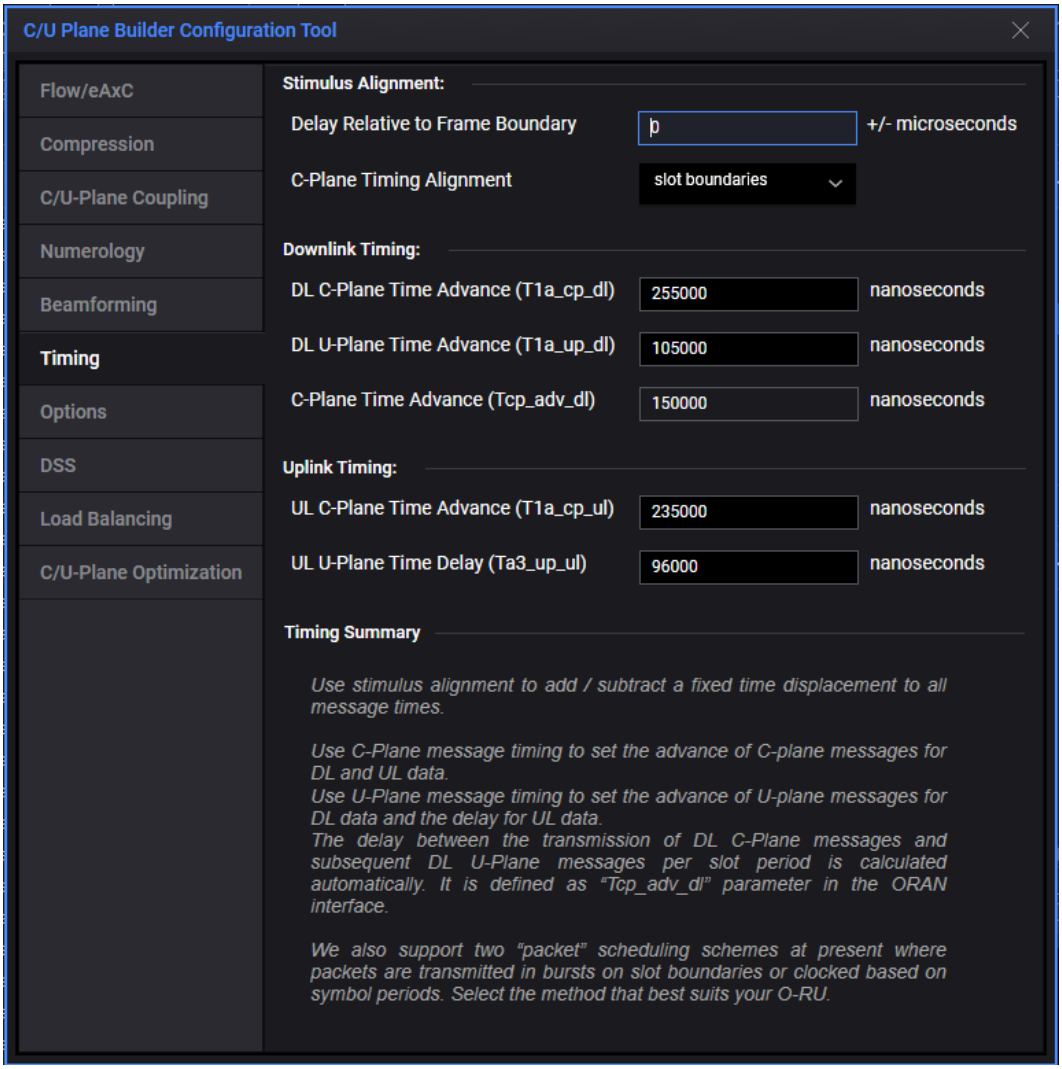
	Earliest transmission from O-DU	Latest Transmission from O-DU
U-Plane	T1a_max_up ≤ T2a_max_up + T12_min	T1a_min_up ≥ T2a_min_up + T12_max
C-Plane	T1a_max_cp_dl ≤ T2a_max_cp_dl + T12_min	T1a_min_cp_dl ≥ T2a_min_cp_dl + T12_max

\

Downlink			Uplink	
Parameter	**µ sec**		**Parameter**	**µ sec**
T1a_max_up	206		Ta3_max	150
T1a_min_up	135		Ta3_min	40
T2a_max_up	206		Ta4_max	205
T2a_min_up	80		Ta4_min	40
T12_max	55		T34_max	55
T12_min	0		T34_min	0

Inside Range testing

Configure the Uplink and Downlink timing within the range defined in Table 5-8 **Error! Reference source not found.** and check ORAN IP should not receive Late or Early packets.



Early testing

Configure the Uplink and Downlink timing before the range defined in Table 5-8**Error! Reference source not found.** and check all packets are dropped in ORAN IP should updating Early packet counters.

Late Testing

Configure the Uplink and Downlink timing after the range defined in **Error! Reference source not found.** and check all packets are dropped in ORAN IP should updating Late packet counters.

5.5 VLAN classification of M-Plane

5.6 FDV buffer

ORAN packets are sent out of order with respect to eAxC numbering and fragmented packet for a specific eAxC. The FDV is tested to ensure packets are put in the correct order in the FDV buffer.

ORAN packets with advanced scheduling from Keysight Oran Studio are sent to Sail River O-RU to verify FDV can absorb the Fronthaul delay in the received packets. 10 km Fronthaul delay verification is outside of the Sail River VTP, and may be verified by the system integrator.

5.7 DL/UL Phase Compensation

Phase compensation module is sanity checked in MS1.5. Two center frequency values

for Phase Compensation are tested in MS1.5:

1- DL and UL center carrier frequency 3.7497 GHz
2- DL and UL center carrier frequency 3.950 GHz

The phase compensation module is configured for both UL and DL using the given center frequency values. The complex phasor generated through the phase compensation module is tested against the expected values for one Symbol.

This testing can be done by enabling and disabling phase compensation and comparing the phase rotation at the output of the iFFT in TD. Similar way in UL however the comparison is done in FD.

5.8 Lower PHY Freq Domain Capture (IFFT_IN or FFT_OUT)

Sail River supports multiple bandwidths, and fixed sampling rate of 122.88MHz. The Capture Size of IFFT Input /FFT output signal is 3276, which is the 1 symbol data of an AxC.

1 symbol has 273 PRB and 1 PRB has 12 resource elements = 3276 Samples

- Each symbol will be captured and plotted.
- The captured data will be compared with the expected data and EVM and NMSE are calculated.

5.9 Lower PHY Time Domain Capture (IFFT OUT or FFT IN)

Sail River supports multiple bandwidths (60 and 100 MHz for phase II), and fixed sampling rate of 122.88MHz, corresponding to Symbols size of 4096 samples in time domain.

In the IFFT_OUT/FFT IN signal, the capture size of 0th symbol of an AxC is 4448 and the capture size of remaining symbols in an AxC is 4384.

- Each symbol will be captured and plotted.

- The captured data will be compared with the expected data and EVM and NMSE are reported.

5.10 DUC/Channel Filter OUT

- The capture size of DUC OUT for symbol 0 & 14 is 17792 and all other symbols have capture size of 17536 samples.
- The capture can be done for each AxC using a multiplexer.
- The Captured data will be compared with their respective expected data. The EVM is calculated for every capture.

5.11 DDC IN

- The capture size of DDC IN for symbol 0 & 14 is 17792 and all other symbols have capture size of 17536 samples.
- The capture can be done for each AxC through a multiplexer.
- The Captured data will be compared with their respective expected data. The EVM is calculated for every capture to get required EVM.

5.12 PRACH and UL Arbiter Validation

Long sequence PRACH formats F0 are tested using the PRACH buffer and verified at the output of the PRACH by detecting the Preamble ID and the timeOffset in Matlab. The PRACH payload dumped in FPGA as well as in Keysight Oran Studio and verified by Matlab 5GNR PRACH application toolbox.

PRACH and PUSCH packets header and payload are captured and analysed in Keysight Oran Studio to ensure the time reference and arbitration of the received packets for both PRACH and PUSCH are as expected.

Short Preamble PRACH testing for MS1.5:

Short Format PRACH is tested in format B4, configuration index 156, and with the following offset frequencies: -3276, -3252, -3228, -3204, -3180, -3156, -3132, -3108, -3084, and -3060.

Long Preamble PRACH testing for MS1.5:

The long PRACH is tested for Format 0 with different config index, Frequency offset, Preamble ID and sequence number in both M_plane and C-plane configuration.

The test vectors are generated using Matlab 5G toolbox for a particular root sequence (e.g. 53) and a Preamble ID (e.g. 23) and stored in the PRACH pattern buffer. The data is passed through the short/long PRACH IP and captured/detected in Keysight VSA.

Proposed Long PRACH index for testing: 0, 16, 21, and 27.

PRACH Configuration Index	Preamble format	$n_{SFN} \bmod x = y$		Subframe number	Starting symbol	Number of PRACH slots within a subframe	$N_t^{RA,slot}$, number of time-domain PRACH occasions within a PRACH slot	N_{dur}^{RA}, PRACH duration
		x	y					
0	0	16	1	9	0	-	-	0
1	0	8	1	9	0	-	-	0
2	0	4	1	9	0	-	-	0
3	0	2	0	9	0	-	-	0
4	0	2	1	9	0	-	-	0
5	0	2	0	4	0	-	-	0
6	0	2	1	4	0	-	-	0
7	0	1	0	9	0	-	-	0
8	0	1	0	8	0	-	-	0
9	0	1	0	7	0	-	-	0
10	0	1	0	6	0	-	-	0
11	0	1	0	5	0	-	-	0
12	0	1	0	4	0	-	-	0
13	0	1	0	3	0	-	-	0
14	0	1	0	2	0	-	-	0
15	0	1	0	1,6	0			0
16	0	1	0	1,6	7	-	-	0
17	0	1	0	4,9	0	-	-	0
18	0	1	0	3,8	0	-	-	0
19	0	1	0	2,7	0	-	-	0
20	0	1	0	8,9	0	-	-	0
21	0	1	0	4,8,9	0	-	-	0
22	0	1	0	3,4,9	0	-	-	0
23	0	1	0	7,8,9	0	-	-	0
24	0	1	0	3,4,8,9	0	-	-	0
25	0	1	0	6,7,8,9	0	-	-	0
26	0	1	0	1,4,6,9	0	-	-	0
27	0	1	0	1,3,5,7,9	0	-	-	0
28	1	16	1	7	0	-	-	0

The Frame ID calculation is done in HPS based on Time of the Day (ToD), as described in ORAN specification for SFN number generation. These calculations are dumped into a spreadsheet and checked for a period of 10 Radio Frames.

5.13 M-plane Sanity Check through 25GbE

The HPS in Sail River design is expected to send and receive M-Plane traffic through the front-end 25GbE HSSI based Ethernet link and the mSGDMA data path. The ORAN M-Plane software stack is outside of scope of Sail River. M-plane runs on models over SSH and the bidirectional traffic is over L3 (TCP). The sanity tests on the O-DU and the O-RU (i.e. HPS) will ensure that the HPS can send and receive TCP packets through available traffic tools (e.g. iperf3) to prove the M-Plane readiness.

The M-plane Server application runs on user space of Linux Server. The application uses TCP/IP network protocol stack for communication. The kernel has a network driver which manages the data transfer between application and physical interface. M-plane packets are sent directly through the physical interface. IP address must be assigned for the specific interface for this communication.

The M-plane client application runs on HPS user space of Hitek board. The application uses TCP/IP network protocol stack for communication. The kernel space has the Intel HSSI driver and once the driver is loaded the physical interface can be detected in HPS with ifconfig command. The SFP28 interface is used for M-plane. After the interface is listed with ifconfig command, an IP address must be assigned. The Hardware part has the SFP28 physical interface, MAC, PHY and 1588 PTP subsystem in FPGA. Once the M-plane packets are received in physical interface, it is sent to DMA subsystem through

PHY and MAC. The HSSI driver in kernel space initializes PHY & MAC and mSGDMA for packet handling between HPS and physical interface.

The hardware setup diagram for M-plane testing refer Figure 5-5

Figure 5-5 Hardware setup diagram for M-plane

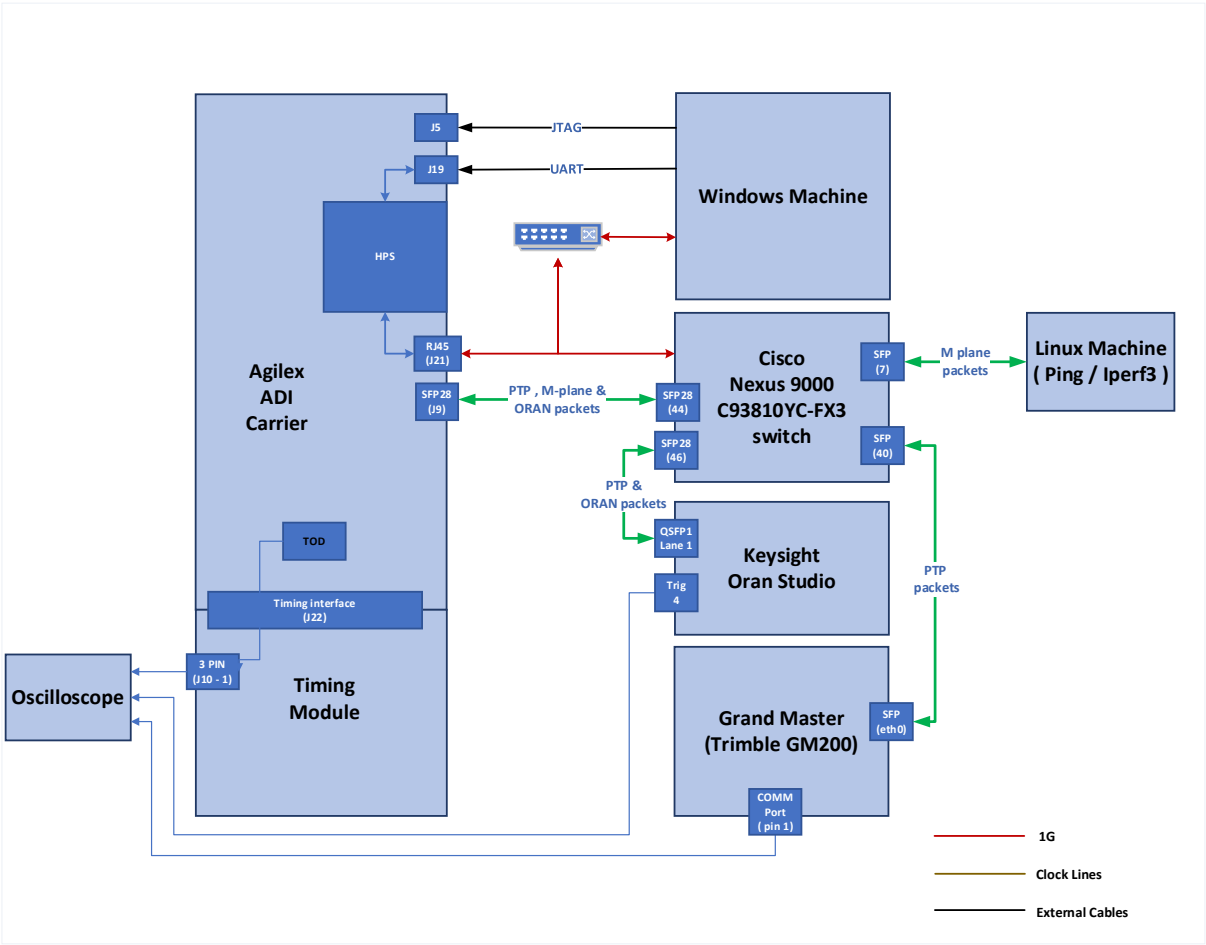
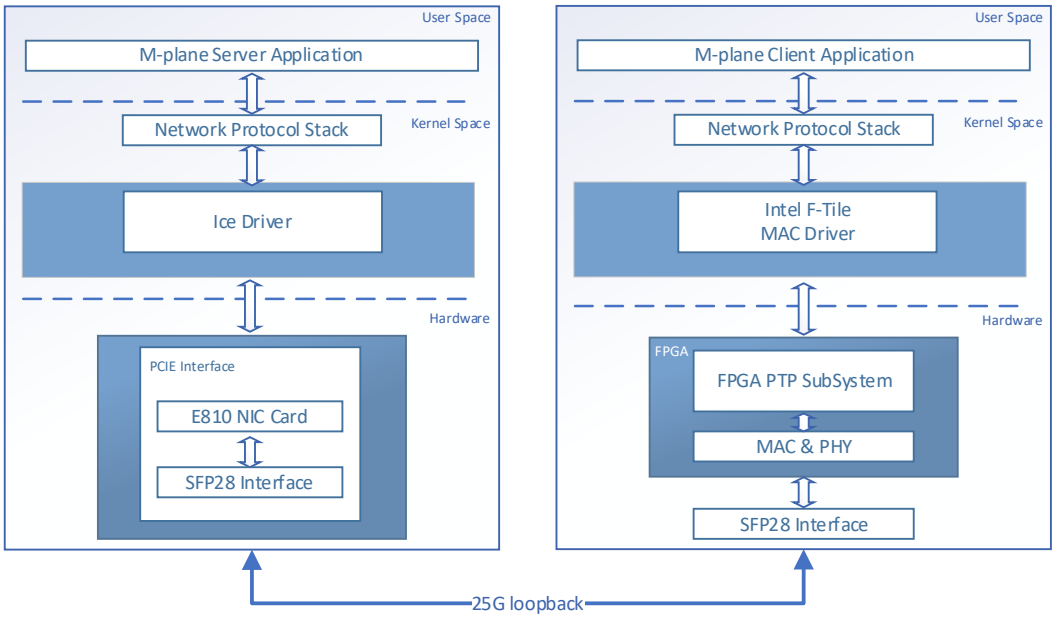


Figure 5-6 Software Architecture for M-plane



The M-plane Server Application opens the socket connection, binds with the server address and port number, and listens for the client. The M-plane client application creates a socket and connect with server IP address and same port number to accept the connection in server. Once the connection is established Packets can be send and receive between server and client.

The IP packets is framed with the payload of packet header and command header with the configurable data and sent from (Linux server). The client application running in HPS userspace of O-RU(Hitek) will be waiting to receive the packets from Linux server. The Packet has an IP Header, Packet Header, Command Header following the data.

The below table has the Header structure for the M-plane packets following the IP packet header.

Table 5-9 M-plane header structure

Headers	Field	Bits		Description
Packet Header	Header_String	24		This is the first field of Packet. Incoming packets will be checked for this string
	Id	8		Unique Id for each incoming command
	Length	32		This includes Command and data
	server_index	8		
	pData	8		Points to the memory that has command header
Command Header	Transfer_Type	1		This field specifies whether command is polling or service type.

Headers	Field	Bits		Description
	Module_Index	7		Index of module to configure.
	Operation_Type	4		Operation to be performed e.g Read, Write and Config.
	Instance	4		Instance of module to configure
	Resource	8		Index of the resource to be accessed
	Address	16		Offset address or index of configuration to be done
	Bit_Mask	32		To mask register data during write and read
	Length	22		Number of data offsets succeeding command header
	is_CPRI_CM	1		Decides if it is C&M packet
	server_index	1		
	reserved	24		
	pData	8		Points to the memory that has data

The command header has Module Index, Operation type, Resource Index, Address and bits. The command decoder in client application decodes the packets based on the above indexes. Based on the decoded parameters the command manager will configure the data to the respective registers.

M-plane testing can be performed with ping and iperf3 applications.

Execute the below command for M-plane testing with ping application.

$ ping -I <interface name> -s <Packet size> -i <wait> <Host>

-I <interface name>: - interface used for outgoing IPv4 multicasts.

-s < Packet size >: - Packet size is number of data bytes to be sent. The default is 56, which converted to 64 data bytes when combined with the 8 bytes of header data.

-i < wait >: - Waits the number of seconds specified by the Wait variable between the sending of each packet. The default is to wait for one second between each packet.

Table 5-10 Ping test cases

S.No	Packet size(s)	Wait (i)
1	120	0.001
2	120	0.0001
3	1520	0.5
4	1520	0.1

Execute the below command for M-plane testing with iperf3 application in server mode.

$ ./iperf3 -s

-s:- running in server mode

Execute the below command for M-plane testing with iperf3 application in client mode.

$ ./iperf3 -M <segment size> -b <bandwidth in bits/sec> -c <server IP address >

-c :- running in client mode

-M <segment size>:- maximum TCP segment size (MTU – 40 bytes).

-b < bandwidth in bits/sec >:- Limits the target bandwidth.

-R :- use this option to test the reverse link.

Table 5-11 iperf3 test cases

TCP segment size (Bytes)	MTU size (Bytes)	Measured throughput	
		iperf3 sender (Mbps)	iperf3 receiver (Mbps)
88	128		
128	168		
256	296		
512	552		
1024	1064		
1280	1320		
1460	1500		

5.14 Packet Header and Time Reference Checking

Header time reference generation in UL O-RU is checked in Keysight Oran studio by opening the captured pcap file in explorer tab. The header information is analyzed and verified.

5.15 L2 COS, Data Flow Identification, and Qos

- Test C/U/M plane packets are sent from Keysight Oran studio with different VLAN tags and MAC addresses. The VLAN LUT is also programmed in the eCPRI flow identification module. The correct forwarding of the packet based on VLAN filtering is verified in FPGA and HPS. More information will be provided in the next version of the VTP document.
- The packets after the PCP analysis by eCPRI IP are received in HPS. A sanity check is performed in HPS by using TCP dump for a period of several seconds

to make sure we never lose PTP packets, and they are prioritized over M-plane or any other IP packets.

5.16 Eaxc configurability

ORAN defines a 16 bits eAxC ID to establish an end-point identifier between O-DU and O-RU. The bit allocation is assumed flexible as shown in Figure 5-7

Figure 5-7 eAxC ID bit allocation

Value range: {0000 0000 0000 0000b-1111 1111 1111 1111b = eAxC ID}

Bit allocations:

0 (msb)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15 (lsb)	Number of Octets
DU_Port_ID				BandSector_ID				CC_ID				RU_Port_ID				2

In DL direction 8 different AxCs (4T4R/2CCs) are supported in Sail River for PDSCH and U-plane as PRACH is absent in Downlink. However, O-DUs may use this ID with their own logical identifier values. In UL direction 16 different AxCs (4T4R/2CCs, and 8 PRACH) are supported in Sail River. The Transmission PCAP is generated with random values in PC-ID fields and the same values are configured through M-plane. Refer **Error! Reference source not found.** for the register configuration. A mapper i s implemented to translate the O-DU logical eAxC IDs into O-RU physical eAxCs ID.

5.17 CC Enable/Disable and Mixed Bandwidth configuration

For CC Enable/Disable testcase, we have two registers in radio config module named bw_config_cc1 and bw_config_cc2. In that configure bit 7 to 0/1 to Enable CC/Disable CC. Configure bit 0 to 6 with Value 10 to select bandwidth as 60 MHz or 14 to select bandwidth as 100 MHz. when we enable both CC we will receive all the axc's in ORAN studio. If we disable any one CC we will receive only 4 axc'c of CC which is enabled.

5.18 Frame Sync reset

The Frame Sync block implements a counter-based mechanism to provide a timing reference for the entire O-RU design. This block aligns the counter start to the radio frame wrap-around boundary of 10.24 seconds. This alignment is performed after PTP synchronization has been achieved between the Master clock and O-RU.

For frame resetting, alpha and beta values are included to calculate the frame number based on the following formula:

$$FrameNumber = floor(\frac{GPSseconds - \beta * 0.01 - \frac{\alpha}{1.2288 * 10^9}}{framePeriodinSeconds})mod(maxFrameNumber + 1)$$

*Alpha and Beta are considered to be Zero in phase II Sail River.

Here, alpha and beta values are zero.

The ToD received from PTP adjustment is in TAI time which has an offset compared to GPS second. The TAI to GPS second conversion is required to calculate the next SFN boundary in PTP domain.

Assuming:

PTPsecond: representing the current ToD in PTP domain,

GPSsecond: representing the current ToD in GPS domain,

Offset = 3159964819,

The next roll-over boundary in PTP domain is calculated as follows:

GPSsecond_shifted = GPSsecond – 0.01*β – α/(1.2288*1e9)

Last_rollover_Boundary = floor(GPSsecond_shifted / 10.24) * 10.24

Δ (till next rollover) = 10.24 – (GPSsecond_shifted - Last_rollover_Boundary)

Next rollover boundary in PTP domain = PTPsecond + Δ

Frame Number will roll over for every 10.24 seconds which means it will run from 0 to 1023 radio frames of 10msec.

The HPS implement the above formula and finds the Future GPS Seconds value at which the Frame Number will roll over. This is provided to FPGA comparator module through a register interface. When Tod reaches this future time, it sends a go signal (shown as reset in figure above) to the Frame Sync to start all counters at the boundary of SFN 0.

Frame Number Calculation Examples,

Example Case 1: Alpha = Beta = 0

Frame Number = (GPS Sec * 100) mod 1024

Current ToD/PTP Time : 1695608896

GPS Seconds : 1695608896 – 315964819 = 1,379,644,077

Finding Future GPS Seconds when Frame Number Rolls over :-

Future GPS Seconds : Ceil(1379644077/10.24)*10.24 = 1,379,644,078.08

Future Frame Number : 1,379,644,078.08 * 100 mod 1024 = 0

Future GPS to ToD Time conversion to compare with Master ToD IP

Future GPS to ToD : 1,379,644,078.08 + 315964819 = 1695608897.080

ToD Seconds : 1695608897

ToD Nano Seconds : (1695608897.080 – 1695608897)*10^ 9 = 79999923

5.19 Power Meter

Power meters are provided in the design as a debug feature in DL path – iFFT input (frequency domain), UL path - FFT input (time domain).

Power meters take in IQ Data from the UL/DL path selected and calculate Mean Squared Sum and peak calculation on the incoming IQ data and send it to FSM module. FSM module performs the following function.

1. Get the MSS/Pk data from MSS module.
2. Read the threshold data from shared memory (HPS)
3. Perform Histogram calculation at symbol level
4. Update Shared memory (HPS) of histogram at symbol level
5. Perform Statistic Calculation at symbol level
   a. Minimum MSS,

b. Maximum MSS,
c. Total MSS,
d. Total #Samples
e. Peak Power

6. Update Shared memory (HPS) of Statistics at symbol level
7. Perform Bank Toggle of shared memory after Long Integration Time.

The histogram memory and statistic memory are allocated as shown in the following table

Histogram Memory

Table 5-12 Histogram and Statistic Memory mapping

Sl No	Offset	Signal	Data format	Sym No	AxC
1	0	Count-bin-0	Uint16	0	0 (Ping Mem start)
..	..	..			
32	31	Count-bin-31			
33	32	0		1	
65	64	0		2	
129	128	0		3	
417	32*13	0		13	
449	32*14			0	1
897	2*32*14			0	2
1345	3*32*14			0	3
1793	4*32*14	Reserved			
2049	2048+0	Count-bin-0		0	0 (Pong Mem start)
	...	..			
2080	2048+31	Count-bin-31			
2081	2048+32			1	
2113	2048+64			2	
2177	2048+128			3	
2465	2048+32*13			13	
2497	2048+32*14			0	1
2945	2048+2*32*14			0	2
3393	2048+3*32*14			0	3
3841	2048+4*32*14	Reserved			

Sl No	Offset	Signal	Data format	Sym No	AxC
Statistic Memory					
Sl No	**Offset**	**Signal**	**Data format**	**Sym No**	**AxC**
1	0	Min Ymss	16u13	0	0 (Ping Mem start)
2	1	Max Ymss	16u13		
3	2	Total Mss	32u13		
5	4	Total #samples	Uint32		
7	6	Peak power	16u13		
8	7	Reserved	Uint16		
9	8			1	
105	8*13			13	
113	8*14			0	1
225	2*8*14			0	2
337	3*8*14			0	3
449	4*8*14	Reserved			
513	512+0	Min Ymss		0	0 (Pong Mem start)
514	512+1	Max Ymss			
515	512+2	Total Mss			
517	512+4	Total #samples			
519	512+6	Peak power			
520	512+7	Reserved			
521	512+8			1	
617	512+8*13			13	
625	512+8*14			0	1
737	512+2*8*14			0	2
849	512+3*8*14			0	3
961	512+4*8*14	Reserved			

With the help of GUI following debug activities can be performed.

1. Select power meter – Enable/disable
2. Enter mode – single / continuous run

3. Configure 33-thresholds values
4. Enter Long term integration time
5. Display histogram, max (Ymss) , min(Ymss) , total (Ymss) , #samp(Ymss) , max(Ypeak) based on symbol ,axc, cc
6. Calculate NMSE and NPSE.

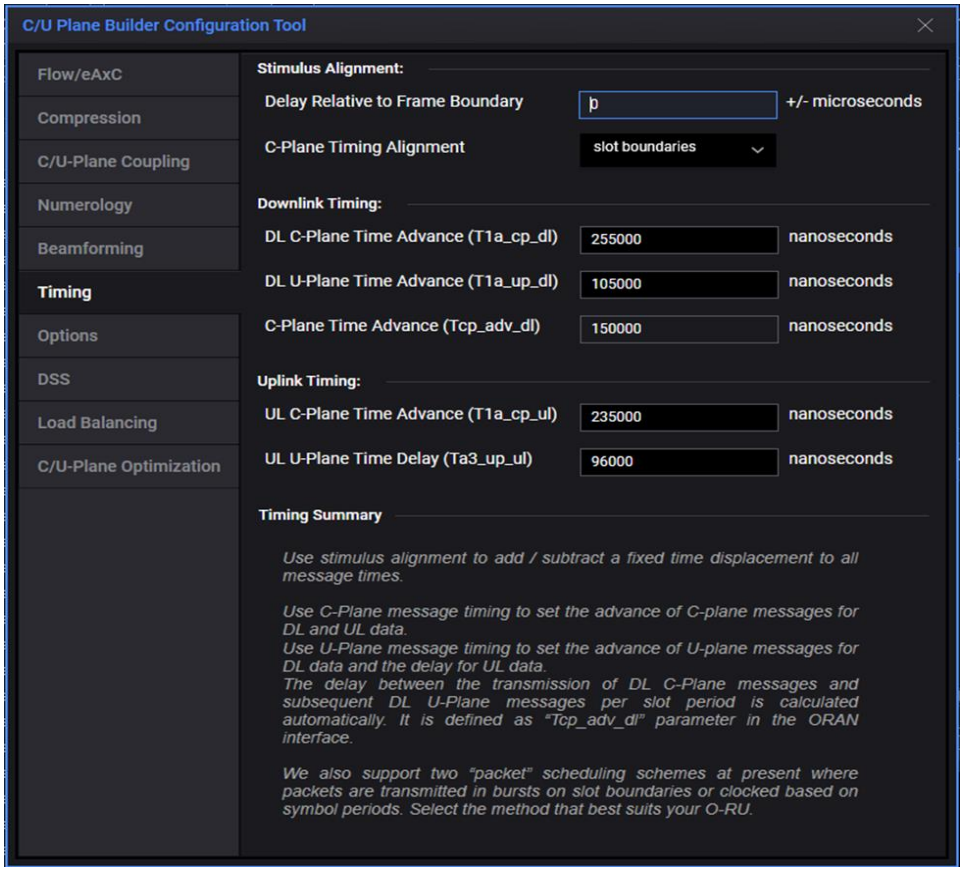
6.0 Keysight Oran Studio advance scheduling

In this section, the test vector generation and scheduling by Keysight Oran studio are described. Keysight Oran studio schedules the O-DU packets (DL) from server to Agilex eSOM.

The following parameters are to be configured in Open RAN studio Application.

- T1a_cp_dl
- T1a_up_dl
- T1a_cp_ul
- T3a_up_ul

Figure 6-1 Advance scheduling in Keysight Oran studio



6.1 Test packet structure for PDSCH

Keysight Oran studio configures 8 AxCs of 60 and 100 MHz BW as supported by the Sail River O-RU. The U-plane packets are generated with 9-Bit Block floating point compression and MTU size configured is 9000 which consists of all max. 273 PRBs in one packet. The timing reference as well as the eCPRI/ORAN header of the U-plane packets are given in the table below.

Table 6-1 Test Packet structure for PDSCH

Seq id	Start prb	Num prb	Axc Id (pc id)	sfn	Sym id	Slot id
0	0	0	0	0	0	0
0	0	0	1	0	0	0
-	-	-	-	-	-	-
-	-	-	-	-	-	-
0	0	0	7	0	0	0
1	0	0	0	0	1	0
1	0	0	1	0	1	0
-	-	-	-	-	-	-
-	-	-	-	-	-	-
1	0	0	7	0	1	0
-	-	-	-	-	-	-
-	-	-	-	-	-	-
13	0	0	0	0	13	0
13	0	0	1	0	13	0
-	-	-	-	-	-	-
-	-	-	-	-	-	-
13	0	0	7	0	13	0
14	0	0	0	0	0	1
14	0	0	1	0	0	1
-	-	-	-	-	-	-
-	-	-	-	-	-	-
14	0	0	7	0	0	1
-	-	-	-	-	-	-
-	-	-	-	-	-	-
27	0	0	0	0	13	1

Seq id	Start prb	Num prb	Axc Id (pc id)	sfn	Sym id	Slot id
27	0	0	1	0	13	1
-	-	-	-	-	-	-
-	-	-	-	-	-	-
27	0	0	7	0	13	1
-	-	-	-	-	-	-
-	-	-	-	-	-	-
0-255			0-7	0-9	0-13	0-1

6.2 Test packet structure for PUSCH

Keysight Oran studio receives 8 AxCs of 100 MHz BW from the Sail River O-RU. The u-plane fragment size is configured by 30 PRBs. For each symbol, 9 first packets include 30 PRBs and the last packet carries 3 PRBs. The timing reference as well as the eCPRI/ORAN header of the U-plane packets are given in the table below.

Table 6-2 Test Packet structure for PUSCH

Seq id	Start prb	Num prb	Axc Id (pc id)	sfn	Sym id	Slot id
0	0	30	0	0	0	0
1	30	30	0	0	0	0
-	-	-	-	-	-	-
-	-	-	-	-	-	-
8	240	30	0	0	0	0
9	270	3	0	0	0	0
-	-	-	-	-	-	-
-	-	-	-	-	-	-
0	0	30	7	0	0	0
1	30	30	7	0	0	0
-	-	-	-	-	-	-
-	-	-	-	-	-	-
8	240	30	7	0	0	0
9	270	3	7	0	0	0
130	0	30	0	0	13	0
131	30	30	0	0	13	0
-	-	-	-	-	-	-

Seq id	Start prb	Num prb	Axc Id (pc id)	sfn	Sym id	Slot id
-	-	-	-	-	-	-
138	240	30	0	0	13	0
139	270	3	0	0	13	0
-	-	-	-	-	-	-
-	-	-	-	-	-	-
130	0	30	7	0	13	0
131	30	30	7	0	13	0
-	-	-	-	-	-	-
-	-	-	-	-	-	-
138	240	30	7	0	13	0
139	270	3	7	0	13	0
140	0	30	0	0	0	1
141	30	30	0	0	0	1
-	-	-	-	-	-	-
-	-	-	-	0	-	-
148	240	30	0	0	0	1
149	270	3	0	0	0	1
-	-	-	-	-	-	-
-	-	-	-	-	-	-
140	0	30	7	0	0	1
141	30	30	7	0	0	1
-	-	-	-	-	-	-
-	-	-	-	-	-	-
148	240	30	7	0	0	1
149	270	3	7	0	0	1
-	-	-	-	-	-	-
-	-	-	-	-	-	-
-	-	-	-	-	-	-
0-255			0-7	0-9	0-13	0-1

6.3 Test packet structure for PRACH

Table 6-3 PRACH Packet structure (Format-0 CFG 27)

Seq id	Start prb	Num prb	Axc Id	sfn	sym	Slot id
0	0	72	8	1	0	0
0	0	72	9	1	0	0
0	0	72	…	1	0	0
0	0	72	…	1	0	0
0	0	72	14	1	0	0
0	0	72	15	1	0	0
1	0	72	8	3	0	0
1	0	72	9	3	0	0
1	0	72	…	3	0	0
1	0	72	…	3	0	0
1	0	72	14	3	0	0
1	0	72	15	3	0	0
2	0	72	8	5	0	0
2	0	72	9	5	0	0
2	0	72	…	5	0	0
2	0	72	…	5	0	0
2	0	72	14	5	0	0
2	0	72	15	5	0	0
3	0	72	8	7	0	0
3	0	72	9	7	0	0
3	0	72	…	7	0	0
3	0	72	…	7	0	0
3	0	72	14	7	0	0
3	0	72	15	7	0	0
4	0	72	8	9	0	0
4	0	72	9	9	0	0
4	0	72	…	9	0	0
4	0	72	…	9	0	0

Seq id	Start prb	Num prb	Axc Id	sfn	sym	Slot id
4	0	72	14	9	0	0
4	0	72	15	9	0	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
0-255	0	72	8-15	0-9	0/7 *	0

Keysight Oran studio receives 8 AxCs of PRACH preambles from the Sail River O-RU. The PRACH packets are transmitted with 72 PRBs per packet. The timing reference as well as the eCPRI/ORAN header of the PRACH packets are given in Table 6-3

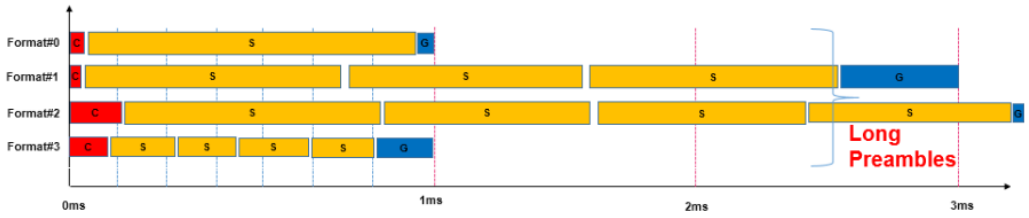
Symbol number can be either 0 or 7 based on the PRACH configuration index

7.0 PRACH Testing

For PRACH, below given different Long-Preamble sequence, Sail River supports only Format-0.

Format	L_{RA}	Δf^{RA}	N_u	N_{CP}^{RA}	Support for restricted sets
0	839	1.25 kHz	24576κ	3168κ	Type A, Type B
1	839	1.25 kHz	$2 \cdot 24576\kappa$	21024κ	Type A, Type B
2	839	1.25 kHz	$4 \cdot 24576\kappa$	4688κ	Type A, Type B
3	839	5 kHz	$4 \cdot 6144\kappa$	3168κ	Type A, Type B

Table 7-1 PRACH Long Preamble duration for all Long Formats



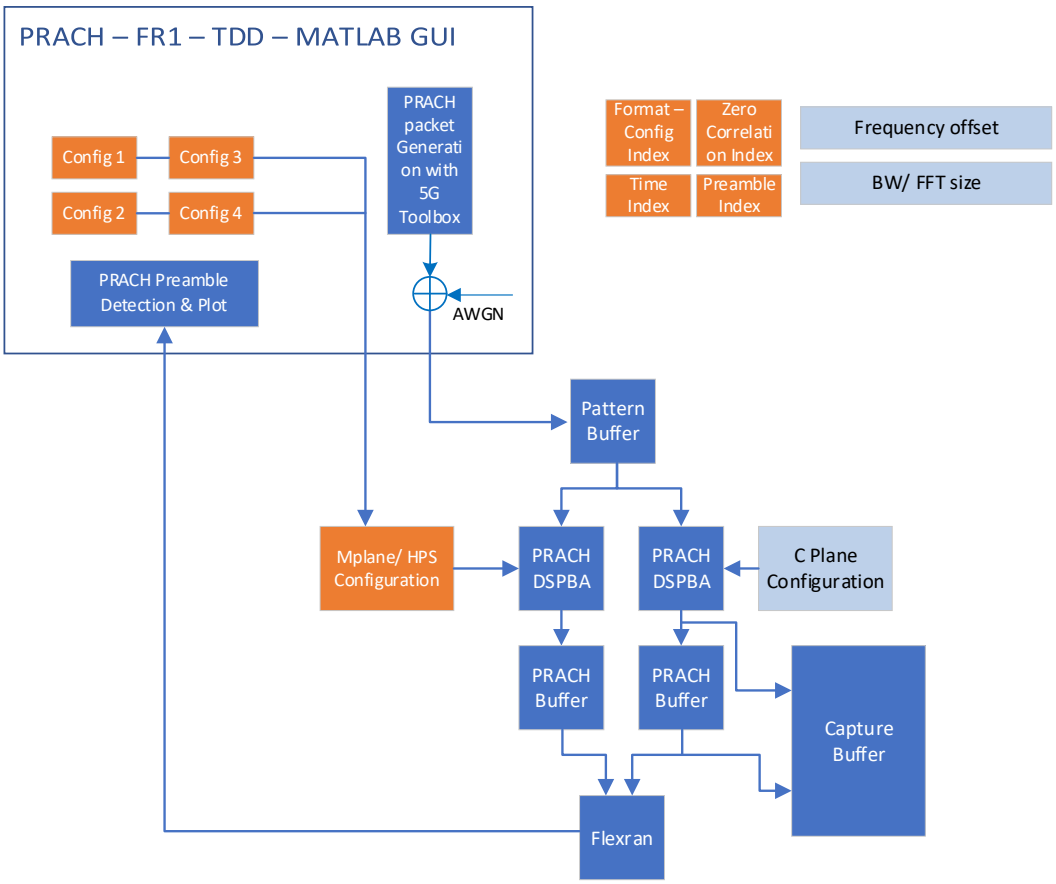
Test Vectors are generated for long preamble format-0 from MATLAB 5G Toolbox in run time based on the format we select from MATLAB GUI. Once Test vectors are generated this will be loaded into FPGA Pattern buffer. AWGN noise can be added to the PRACH test waveform for performance evaluation.

C plane information such as Frequency offset and BW/FFT Size will be received from C plane packet from O-DU Emulator. All M-Plane/HPS Configuration will be configured from MATLAB GUI and HPS register map Configures that into PRACH DSPBA IP. Refer Figure 7-1 for the registers that are configured.

DSPBA IP output will be captured by capture buffer and HPS reads this and transfers to MATLAB. This will be compared against the PRACH MATLAB Model.

The same PRACH data will be sent to Keysight Oran studio after eCPRI/ORAN packet header formation. Keysight Oran studio will receive the packets. In the captured pcap, extract one axc's packets using wireshark , save the pcap file. The extracted pcap file is transferred to MATLAB server and analyzed.

Table 7-2 PRACH Testing procedure for different preamble format



Design for Debug Feature

7.1 U Plane Profiling

U Plane Profiling captures the snapshot of incoming U plane packets, which enables the user to validate the incoming Packet header fields and the frame numbers running across the system. This feature helps to ensure whether the system has received all proper U plane packets from O-DU Emulator.

The FPGA design has a buffer structure to store the packet headers of specified length. In addition to packet headers, other timing information will also be logged into the buffer including the Frame Sync timers (current RFN/SFN/Sym info) and ToD timers (GPS time).

This buffer is mapped to HPS h2f Bridge for Burst of data transfer b/w FPGA to HPS. The SoC application is responsible for starting and stopping the packet dumping into FPGA. Then the application reads the FIFO data, decodes the required Header info and writes it into the .csv file. From that file, an offline analysis will be done to validate the log.

- From the sequence of packet headers, the fields including PC ID, Frame id, Sub frame ID, start PRB, Num PRB, Section ID, Slot ID, Sym ID etc. will be verified.
- From frame sync timers, the frame numbers like SFN, RFN, symbol number and sub carrier number will be verified.
- From TOD, we will get seconds and fractional nanoseconds information.

The Incoming U plane packet will have the below packet structure along with Ethernet and eCPRI/ORAN packet header.

8.0 Design for Debug Feature

8.1 U Plane Profiling

U Plane Profiling captures the snapshot of incoming U plane packets, which enables the user to validate the incoming Packet header fields and the frame numbers running across the system. This feature helps to ensure whether the system has received all proper U plane packets from O-DU Emulator.

The FPGA design has a buffer structure to store the packet headers of specified length. In addition to packet headers, other timing information will also be logged into the buffer including the Frame Sync timers (current RFN/SFN/Sym info) and ToD timers (GPS time).

This buffer is mapped to HPS h2f Bridge for Burst of data transfer b/w FPGA to HPS. The SoC application is responsible for starting and stopping the packet dumping into FPGA. Then the application reads the FIFO data, decodes the required Header info and writes it into the .csv file. From that file, an offline analysis will be done to validate the log.

- From the sequence of packet headers, the fields including PC ID, Frame id, Sub frame ID, start PRB, Num PRB, Section ID, Slot ID, Sym ID etc. will be verified.
- From frame sync timers, the frame numbers like SFN, RFN, symbol number and sub carrier number will be verified.
- From TOD, we will get seconds and fractional nanoseconds information.

The Incoming U plane packet will have the below packet structure along with Ethernet and eCPRI/ORAN packet header.

8.2 Profiling results exported as CSV file

Figure 8-1 Downlink U plane Profiling CSV File Example

Incoming Packet Header						Timing Reference from Frame Sync Module				Timing Referencefrom ToD IP	
ecpriPcid	frameId	subframeId	startPrbu	numPrbu	symbolId	sample_cnt	sym_cnt	sfn_cnt	rfn_cnt	sec	ns_fns
0	0	0	0	91	0	6657	0	0	0	1620281653	1.37635E+13
1	0	0	0	91	0	6845	0	0	0	1620281653	1.37635E+13
2	0	0	0	91	0	7035	0	0	0	1620281653	1.37635E+13
3	0	0	0	91	0	7223	0	0	0	1620281653	1.37636E+13
0	0	0	91	91	0	7412	0	0	0	1620281653	1.37636E+13
1	0	0	91	91	0	7600	0	0	0	1620281653	1.37636E+13
2	0	0	91	91	0	7790	0	0	0	1620281653	1.37637E+13
3	0	0	91	91	0	7978	0	0	0	1620281653	1.37637E+13
0	0	0	182	91	0	8166	0	0	0	1620281653	1.37637E+13
1	0	0	182	91	0	8355	0	0	0	1620281653	1.37637E+13
2	0	0	182	91	0	8544	0	0	0	1620281653	1.37638E+13
3	0	0	182	91	0	8733	0	0	0	1620281653	1.37638E+13
0	0	0	0	91	0	8921	0	0	0	1620281653	1.37638E+13
1	0	0	0	91	0	9110	0	0	0	1620281653	1.37638E+13
2	0	0	0	91	0	9299	0	0	0	1620281653	1.37639E+13
3	0	0	0	91	0	9488	0	0	0	1620281653	1.37639E+13
0	0	0	91	91	0	9676	0	0	0	1620281653	1.37639E+13
1	0	0	91	91	0	9864	0	0	0	1620281653	1.37639E+13
2	0	0	91	91	0	10054	0	0	0	1620281653	1.3764E+13
3	0	0	91	91	0	10242	0	0	0	1620281653	1.3764E+13

Figure 8-2 Uplink U Plane Profiling CSV File Example

Incoming Packet Header						Timing Reference from Frame Sync Module				Timing Referencefrom ToD IP	
ecpriPcid	frameId	subframeId	startPrbu	numPrbu	symbolId	sample_cnt	sym_cnt	sfn_cnt	rfn_cnt	sec	ns_fns
0	0	0	0	91	0	6657	0	0	0	1620281653	1.37635E+13
8	0	0	0	12	0	6845	0	0	0	1620281653	1.37635E+13
0	0	0	91	91	0	7035	0	0	0	1620281653	1.37635E+13
9	0	0	0	12	0	7223	0	0	0	1620281653	1.37636E+13
0	0	0	182	91	0	7412	0	0	0	1620281653	1.37636E+13
10	0	0	0	12	0	7600	0	0	0	1620281653	1.37636E+13
1	0	0	0	91	0	7790	0	0	0	1620281653	1.37637E+13
11	0	0	0	12	0	7978	0	0	0	1620281653	1.37637E+13
1	0	0	91	91	0	8166	0	0	0	1620281653	1.37637E+13
12	0	0	0	12	0	8355	0	0	0	1620281653	1.37637E+13
1	0	0	182	91	0	8544	0	0	0	1620281653	1.37638E+13
13	0	0	0	12	0	8733	0	0	0	1620281653	1.37638E+13
2	0	0	0	91	0	8921	0	0	0	1620281653	1.37638E+13
14	0	0	0	12	0	9110	0	0	0	1620281653	1.37638E+13
2	0	0	91	91	0	9299	0	0	0	1620281653	1.37639E+13
15	0	0	0	12	0	9488	0	0	0	1620281653	1.37639E+13
2	0	0	182	91	0	9676	0	0	0	1620281653	1.37639E+13
8	0	0	0	25	0	9864	0	0	0	1620281653	1.37639E+13
3	0	0	0	91	0	10054	0	0	0	1620281653	1.3764E+13
9	0	0	0	25	0	10242	0	0	0	1620281653	1.3764E+13
3	0	0	91	91	0	10431	0	0	0	1620281653	1.3764E+13
10	0	0	0	25	0	10619	0	0	0	1620281653	1.37641E+13

9.0 Revision History

Date	Version	Changes
2025-05-21	0.0.1	Gen3 Initial Release