Lattice ORAN 1.1 IEEE 1588 - Embedded PTP Programming User Guide

User Guide

FPGA-UG-02203-1.0

January 2024

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS, with all faults, and all associated risk is the responsibility entirely of the Buyer. The information provided herein is for informational purposes only and may contain technical inaccuracies or omissions, and may be otherwise rendered inaccurate for many reasons, and Lattice assumes no obligation to update or otherwise correct or revise this information. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. LATTICE PRODUCTS AND SERVICES ARE NOT DESIGNED, MANUFACTURED, OR TESTED FOR USE IN LIFE OR SAFETY CRITICAL SYSTEMS, HAZARDOUS ENVIRONMENTS, OR ANY OTHER ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE, INCLUDING ANY APPLICATION IN WHICH THE FAILURE OF THE PRODUCT OR SERVICE COULD LEAD TO DEATH, PERSONAL INJURY, SEVERE PROPERTY DAMAGE OR ENVIRONMENTAL HARM (COLLECTIVELY, "HIGH-RISK USES"). FURTHER, BUYER MUST TAKE PRUDENT STEPS TO PROTECT AGAINST PRODUCT AND SERVICE FAILURES, INCLUDING PROVIDING APPROPRIATE REDUDANCIES, FAIL-SAFE FEATURES, AND/OR SHUT-DOWN MECHANISMS. LATTICE EXPRESSLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS OF THE PRODUCTS OR SERVICES FOR HIGH-RISK USES. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Contents

Figures

Tables

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
BC	Boundary Clock
DUT	Device Under Test
E2E	End to End Delay Measurement
GNSS	Global Navigation Satellite System (Clock source from GNSS Chip)
GUI	Graphical User Interface
IPv4	TCP/IP Protocol Suit version 4
IPv6	TCP/IP Protocol Suit version 6
OC	Ordinary Clock in IEEE1588-2019
OCXO	Oven Controlled Crystal Oscillators
P2P	Peer to Peer Delay Measurement
PM	Packet Master
SOOC	Slave only Ordinary Clock
SyncE	Synchronous Ethernet
TC	Transparent Clock in IEEE1588-2019
UDP	User Datagram Protocol

1. Embedded PTP Solution Overview

The embedded PTP solution contains FPGA IPs for timestamping units, packet parsers, and a RISC-V processor core. The RISC-V processor core is running a PTP engine along with a custom network stack. This network stack is designed to handle PTP packets. The RISC-V is connected to a UART IP, which provides a communication port between the host machine and hardware. Section 2 and Section 3 provide information on how to communicate with the hardware.

1.1. Features

The current implementation contains several features enabling users to set up, monitor, and control the PTP clock. A detailed description of how to do this is provided in Section 2 and Section 3.

1.2. PTP Options

All supported PTP options are given in Table 1.1. To set these options from middleware, a configuration file can be used (a sample config file is provided in Figure 3.1. Configuration for T-BC ITU-T G.8275.1 Profile).

Table 1.1. PTP Options

SN	Attribute Type	Attribute Particular	Test Clock	Test Profile
1	PTP Node Type	BC	T-BC	ITU-T G.8275.1
2		OC	T-GM	ITU-T G.8275.1
3		TC-E2E	T-TC	ITU-T G.8275.1
4	PTP Message Transport Mechanism	L2	T-BC	ITU-T G.8275.1
5		UDP over IPv4	BC	Default Profile
6		UDP over IPv6	BC	Default Profile
7	PTP Message Transport Routing Mechanism	Multicast over L2	T-BC	ITU-T G.8275.1
8		Multicast over UDP IPv4	BC	Default Profile
9		Multicast over UDP IPv6	BC	Default Profile
10		Unicast over UPD IPv4	PM	ITU-T G.8265.1
11		Unicast over UPD IPv6	PM	ITU-T G.8265.1
12	Best Master Clock Selection	BMCA	OC, BC, GM	Default Profile
13		ABMCA for ITU-T G.8265.1	PM, SOOC	ITU-T G.8265.1
14		ABMCA for ITU-T G.8275.1	T-GM, T-BC, T-TC, T-TSC	ITU-T G.8275.1
15		ABMCA for ITU-T G.8275.2	T-GM-P, T-BC-P, T-TSC-P	ITU-T G.8275.2
16	Path Delay Measurement	E2E	T-TC	ITU-T G.8275.1
17		P2P	Bridge	IEEE802.1AS
18	Clock Behaviour	Two Step Clock	T-BC	ITU-T G.8275.1
19		One Step Clock	T-BC	ITU-T G.8275.1
20	Message Rate	16 Msg/Sec	T-BC	ITU-T G.8275.1
21		32 Msg/Sec	BC	Default Profile
22	PTP Domain Number	As per profile	All Clocks	All profile
23	PTP Clock Class	All profile	All Clocks	All profile
24	PTP Priority1	All profile	All Clocks	All profile
25	PTP Priority2	All profile	All Clocks	All profile
26	Slave Only	All profile	All Clocks	All profile
27	Master Only	All profile	All Clocks	All profile

Table 1.2. Default PTP Options

Data Set	Configurations	Default Value	Range
Default Data Set	twoStepFlag	1	0-1
	clientOnly	0	0-1
	socket_priority	0	0-15
	priority1	128	0-255
	priority2	128	0-255
	domainNumber	0	Different for every profile
	utc_offset	37	
	clockClass	248	0-255
	clockAccuracy	0xFE	
	offsetScaledLogVariance	0xFFFF	Fixed
	free_running	0	0-1
	freq_est_interval	1	
	dscp_event	0	
	dscp_general	0	
	dataset_comparison	ieee1588	IEEE1588 / G.8275.x
	G.8275.defaultDS.localPriority	128	Fixed
	maxStepsRemoved	255	Fixed
Port Data Set	logAnnounceInterval	1	-7 to 4
	logSyncInterval	0	-7 to 4
	operLogSyncInterval	0	
	logMinDelayReqInterval	0	-7 to 4
	logMinPdelayReqInterval	0	-7 to 4
	operLogPdelayReqInterval	0	
	announceReceiptTimeout	3	
	syncReceiptTimeout	0	0-1
	delay_response_timeout	0	
	delayAsymmetry	0	
	fault_reset_interval	4	
	neighborPropDelayThresh	20000000	
	serverOnly	0	0-1
	G.8275.portDS.localPriority	128	Fixed
	asCapable	auto	True/ False/ Auto
	BMCA	ptp	noop / ptp
	inhibit_announce	0	0-1
	inhibit_delay_req	0	0-1
	ignore_source_id	0	0-1

Data Set	Configurations	Default Value	Range
Run time options	assume_two_step	0	0-1
	logging_level	6	Fixed
	path_trace_enabled	0	0-1
	follow_up_info	0	0-1
	hybrid_e2e	0	0-1
	inhibit_multicast_service	0	0-1
	net_sync_monitor	0	0-1
	tc_spanning_tree	0	0-1
	tx_timestamp_timeout	10	Fixed
	unicast_listen	0	
	unicast_master_table	0	
	unicast_req_duration	3600	
	use_syslog	1	
	verbose	0	0-1
	summary_interval	0	
	kernel_leap	1	0-1
	check_fup_sync	0	0-1
	clock_class_threshold	248	Fixed
Servo Options	pi_proportional_const	0	
	pi_integral_const	0	
	pi_proportional_scale	0	
	pi_proportional_exponent	-0.3	
	pi_proportional_norm_max	0.7	
	pi_integral_scale	0	
	pi_integral_exponent	0.4	
	pi_integral_norm_max	0.3	
	step_threshold	0	
	first_step_threshold	0.00002	
	max_frequency	900000000	0-1000000000 (0%-100%)
	clock_servo	pi	pi / linreg / ntpshm / null
	sanity_freq_limit	200000000	0-1000000000 (0%-100%)
	ntpshm_segment	0	
	msg_interval_request	0	0-1
	servo_num_offset_values	10	
	servo_offset_threshold	0	0-1
	write_phase_mode	0	0-1

Data Set	Configurations	Default Value	Range
Transport options	transportSpecific	0x0	
	ptp_dst_mac	01:1B:19:00:00:00	Fixed
	p2p_dst_mac	01:80:C2:00:00:0E	Fixed
	udp_ttl	1	Fixed
	udp6_scope	0x0E	Fixed
	uds_address	/var/run/ptp4l	Fixed
	uds_ro_address	/var/run/ptp4lro	Fixed
Default interface options	clock_type	OC	OC/BC/E2E_TC/P2P_TC
	network_transport	UDPv4	L2/UDPv4/UDPv6
	delay_mechanism	E2E	E2E/P2P/Auto
	time_stamping	hardware	Software / Hardware
	tsproc_mode	filter	filter / raw / filter_weight / raw_weigh
	delay_filter	moving_median	moving_average / moving_median
	delay_filter_length	10	
	egressLatency	0	
	ingressLatency	0	
	boundary_clock_jbod	0	0-1
Clock description	productDescription	;;	Format
	revisionData	;;	Format
	manufacturerIdentity	0:00:00	
	userDescription	;	Format
	timeSource	0xA0	

1.3. PTP Profiles

Supported PTP profiles are as follows:

1. IEEE 1588-2019 default profile
2. ITU-T Telecom profile for frequency synchronization (G.8265.1)
3. ITU-T Telecom profile for phase/time synchronization with full timing support from the network (G.8275.1)
4. ITU-T Telecom profile for phase/time synchronization with partial timing support from the network (G.8275.2)
5. IEEE 802.1AS Timing and Synchronization for Time-Sensitive Applications (gPTP)

1.4. Other Options

Other supported options are listed in Table 1.3.

Table 1.3. Default PTP Options

SN	Option	Comment
1	TC E2E (FPGA Only)	When the software selects the TC as E2E TC, the FPGA take care of all the messages and software doesn't have any control over it.
2	SyncE	The DPLL clock source selection as syncE recovered clock from either port0 or port1.
3	DPLL Clock Source from GNSS	The DPLL clock source selection as GNSS chip.
4	VLAN	No VLAN, with 1 tag and with 2 tags support for VLAN.
5	Delay Compensations	The ingress, egress and PPS delay compensation can be set.
6	GNSS Synchronization	A command can be sent to take the time from the GNSS chip connected at the Hardware and able to synchronize the system clock with GNSS.
7	Port Stats	The device will provide the total number of packets transmitted and received from both the ports in response to a corresponding command.
8	Port Addresses	The MAC, IPv4 and IPv6 address can be set for both the ports using the commands.
9	Servo State	Current servo state can be read from the hardware using appropriate command.
10	Hardware Time Display	The hardware time is displayed on the GUI with resolution of 1 second for the user reference.
11	Master port and Grandmaster	The device will provide the information about the parent port and grandmaster clock information in response to appropriate command.
12	Auto Selection of Attributes	The default attributed are already programmed in the PTP engine. User only needs to provide whichever he wants to change. A list of default PTP attributes is provided in Table 1.2.
13	ARP	The system doesn't support for automatic ARPs. Thus, manual ARP entries can be added by using appropriate commands.

2. Programming Using UART

The configuration and monitoring of the clock can be done using the UART. Appendix B provides a full set of frames for communication with the hardware.

2.1. UART Attributes

The attributes for the UART are listed in Table 2.1.

Table 2.1. UART Attributes

S/N	Attribute	Value
1	Data Size	8 bits
2	Parity Bit	None
3	Stop Bit	1 Bit
4	Baud Rate	115200

After writing the full frame to the hardware, the master needs to wait at least 20–50 minutes before reading the received frame from the hardware.

2.2. Setting up and monitoring the clock

The frame format is provided in Table B.1, and the host needs to prepare the frame according to the format. The frame contains the start of the frame delimiter along with the function code. The function codes are provided in Table B.2. Subsequent tables provide the frames for commands, which include hardware setup, monitoring, and control.

Section 3 provides the commands using the middleware program for the tasks to be performed on the hardware. Follow the section to get the information about the function to do and from Table B.3 to Table B.43 provides the frames and their responses from the hardware to setup, control, and monitor the clock and hardware..

2.3. CRC polynomials

An 8 bit CRC is added to every frame while sending the UART frame to the hardware. The CRC polynomial is shown in Equation 2.1.

$$p(x) = x^7 + x^3 + x^2$$

Equation 2.1 CRC Polynomials

At the reception of the UART frame. the hardware also includes the CRC, and it should be checked with the same polynomial.

3. Programming Using Middleware

The middleware can be used to fully program the PTP clock for any profile and any clock intended. Appendix C provides a full set of commands for reference.

3.1. Establishing the Hardware Connection

Here are the steps to establish the hardware connection.

1. Start the Daemon:
   ```
   sudo mw_daemon.
   ```
2. Check the list of connected devices:
   ```
   sudo daemon_config device_list get.
   ```
3. Select the device:
   ```
   sudo daemon_config device_select set <device index>.
   ```
4. Check the connection status:
   ```
   sudo general_config debug get.
   ```
5. Get the HW version:
   ```
   sudo general_config hw_ver_no get.
   ```
6. Get the FW version:
   ```
   sudo general_config fw_ver_no get.
   ```

If we you able to get the version number correctly, it means the device is connected and ready to take new commands.

3.2. Port Address Get and Set

The hardware doesn't have permanent memory to store the set MAC and IP addresses; hence, every time the hardware is restarted, the user needs to set the MAC and IP addresses.

1. Set the Port Address:

- MAC:
  ```
  sudo sfp_ports_config mac_addr set p0/p1.
  ```
- IPv4:
  ```
  sudo sfp_ports_config ipv4_addr set p0/p1.
  ```
- IPv6:
  ```
  sudo sfp_ports_config ipv6_addr set p0/p1.
  ```

2. Get the Port Address:

- MAC:
  ```
  sudo sfp_ports_config mac_addr get p0/p1.
  ```
- IPv4:
  ```
  sudo sfp_ports_config ipv4_addr get p0/p1.
  ```
- IPv6:
  ```
  sudo sfp_ports_config ipv6_addr get p0/p1.
  ```

3.3. Set Hardware Configurations

The following configurations will go into the FPGA registers directly to tell the FPGA what is running on the PTP engine.

1. Set:

a. TC type
- Not TC:
  ```
  sudo ptp_hw_config tc_type set not.
  ```

- E2E:
  ```
  sudo ptp_hw_config tc_type set e2e.
  ```
- P2P:
  ```
  sudo ptp_hw_config tc_type set p2p.
  ```

b. Network transmission
- L2:
  ```
  sudo ptp_hw_config msg_trans set l2.
  ```
- UPDv4:
  ```
  sudo ptp_hw_config msg_trans set udpv4.
  ```
- UDPv6:
  ```
  sudo ptp_hw_config msg_trans set udpv6.
  ```
- gPTP:
  ```
  sudo ptp_hw_config msg_trans set gPTP.
  ```

c. Clock behavior
- One-Step:
  ```
  sudo ptp_hw_config clk_bhr set 1step
  ```
- Two-Step:
  ```
  sudo ptp_hw_config clk_bhr set 2step
  ```

2. Get:
- TC type:
  ```
  sudo ptp_hw_config tc_type get.
  ```
- Network transmission:
  ```
  sudo ptp_hw_config msg_trans get.
  ```
- Clock behavior:
  ```
  sudo ptp_hw_config clk_bhr get.
  ```

3.4. Set the ARP for Unicast Transmission

For those profiles that use unicast, the hardware requires the ARP entries. These ARP entries can be set using the following commands:

1. Set the ARP use command:
   ```
   sudo arp_addr_config set <index> <mac_addr> <ipv4_addr> <ipv6_addr>
   ```
2. To check the ARP entries, use command:
   ```
   sudo arp_addr_config get <index>
   ```

3.5. Set the PTP Configuration file

The PTP engine uses Linux PTP source code. Thus, a Linux PTP configuration file can be directly loaded to run a specific clock using the command sudo engine_config profile set <path of the configuration file>. There is no get command for profile parameters. The middleware will write the configuration and check it after every write. If it encounters any problems, it will report them, and further writing of the configuration will be stopped. In case of failure, the user needs to run the command again.

3.6. PTE Engine Control

The PTP engine can be controlled and monitored as follows:

1. RUN:
```
sudo engine_config status set run.
```
2. STOP:
```
sudo engine_config status set stop.
```
3. Current Status:
```
sudo engine_config status get.
```

3.7. Servo Monitoring

Servo can be monitored using command:
```
sudo servo_config all get.
```

3.8. Other Parameters Monitoring

The commands for monitoring and setting other parameters are as follows:

1. All parameters:
```
sudo servo_config servo_continuous get.
```
2. PTP port stats:
```
sudo sfp_config port_stats get.
```
3. Mux Parameters:
    a. Set:
    - Data_cpu:
    ```
    sudo mux_config port0/1 set data_cpu.
    ```
    - Same port:
    ```
    sudo mux_config port0 /1 set non_ptp_p0/1.
    ```
    - Different port:
    ```
    sudo mux_config port0/1 set non_ptp_p1/0.
    ```
    - User logic:
    ```
    sudo mux_config port0 set user_traffic.
    ```
    b. Get:
    ```
    sudo mux_config port0/1 get.
    ```
4. Path Delays
    a. Set:
    ```
    sudo delay_config all set <egress> <ingress> <pps>.
    ```
    b. Get:
    ```
    sudo delay_config all get.
    ```
5. DPLL Config:
    a. Set:
    - OCXO:
    ```
    sudo dpll_config  clk_src set ocxo.
    ```
    - syncE to ocxo (port 0):
    ```
    sudo dpll_config clk_src set syncE0_ocxo.
    ```
    - syncE to ocxo (port 1):
    ```
    sudo dpll_config clk_src set syncE1_ocxo.
    ```
    - Gnss to ocxo:
    ```
    sudo dpll_config clk_src set gnss_ocxo.
    ```

- Gnss to syncE:
  ```
  sudo dpll_config clk_src set gnss_syncE0_ocxo.
  ```
- Gnss to syncE to ocxo:
  ```
  sudo dpll_config clk_src set gnss_syncE1_ocxo.
  ```

b. Get:
```
sudo dpll_config clk_src get.
```

6. H/W Time
   a. Set:
   ```
   sudo hw_time_config hw set <time>.
   ```
   b. Get:
   ```
   sudo hw_time_config hw get
   ```
7. VLAN Parameters
   a. Set:
   - Disable:
     ```
     sudo vlan_config port0/1 set disable.
     ```
   - Vlan:
     ```
     sudo vlan_config port0/1 set vlan <value>.
     ```
   - QinQ:
     ```
     sudo vlan_config port0/1 set qinq <value> <value>.
     ```

   b. Get:
   ```
   sudo vlan_config port0/1 get.
   ```
8. GNSS parameters
   a. Set:
   - Restart:
     ```
     sudo gnss_config status set restart.
     ```
   - Stop:
     ```
     sudo gnss_config status set stop.
     ```

3.9. Example of Running T-BC with ITU-T G.8275.1 Profile

To setup a T-BC clock with an 8275.1 profile, we require a set or prior information about the setup. This includes a configuration file to be loaded (as shown in Section 3.9.1) and other information explained in further sections.

3.9.1. T-BC of ITU-T G.8275.1 Configuration file

A configuration file is needed in order to run the T-BC under this profile. An example of a configuration file is shown in Figure 3.1. The configuration file only setups the PTP engine; the settings outside the PTP engine needed to be set manually (provided in Section 3.9.2).

3.9.2. Other Hardware Configurations

Some of the information about the clock, if required, should be put outside the PTP engine. This information is as follows:

1. Clock Behavior: This configuration is a one-step clock.
2. Message Transmission: The message transmission in this profile is ethernet.
3. DPLL Clock Source: The clock source for this profile is always syncE.

```
# Profile        ITU-T G.8275.1
# Profile Clock  Telecom Boundary Clock
#Clock in Standard       Boundary Clock
#Port 1 [ltcnic0]

#Port 2 [ltcnic1]
#

[global]

dataset_comparison        G.8275.x
G.8275.defaultDS.localPriority   128
clock_type       BC
time_stamping    hardware
twoStepFlag      0
clockClass       248
priority1        128
priority2        128
maxStepsRemoved  255
network_transport        L2
logAnnounceInterval      -3
logSyncInterval  -4
logMinDelayReqInterval   -4
serverOnly       0
clientOnly       0
G.8275.portDS.localPriority      128
ptp_dst_mac       01:1B:19:00:00:00
domainNumber     24
#free_running    0
egressLatency    280
ingressLatency   280
[ltcnic0]

boundary_clock_jbod      1
[ltcnic1]

boundary_clock_jbod      1
```

Figure 3.1. Configuration for T-BC ITU-T G.8275.1 Profile

3.9.3. Steps for Clock Setup

Here are the commands for setting up the clock.

1. Start the Daemon using the command sudo mw_daemon.
2. List the connected USB devices using the command shown in Figure 3.2. The commands list out the available devices with their index numbers.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo daemon_config device_list get
List of found devices:
ttyUSB1,0
ttyUSB0,1
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.2. List of connected devices

3. Select the ttyUSB1 device using the command shown in Figure 3.3.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo daemon_config device_select set 0
The device for communication is now set to /dev/ttyUSB1
1, Execution done successfully.
```

Figure 3.3. Device Selection

4. Once the device is selected, the user can check if the connection is established or not by running the command shown in Figure 3.4.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo general_config debug get
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.4. Connection Check

5. To check the FPGA version of the installed bit file, run the command shown in Figure 3.5.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo general_config hw_ver_no get
0.0.21
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.5. FPGA version

6. To check the firmware version of the installed bit file, run the command shown in Figure 3.6.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo general_config fw_ver_no get
1.8.25
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.6. Firmware version

7. Initially we must set the MAC address, IPv4 address and IPv6 address. The command to add the addresses is shown in Figure 3.7, Figure 3.8, and Figure 3.9.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo sfp_ports_config mac_addr set p0 06:01:02:03:04:02
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$ sudo sfp_ports_config mac_addr get p0
06:01:02:03:04:02
1, Execution done successfully.
```

Figure 3.7. Setting the MAC address

```
ptp8@ptp8-OptiPlex-5050:~$ sudo sfp_ports_config ipv4_addr set p0 192.168.3.20
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$ sudo sfp_ports_config ipv4_addr get p0
192.168.003.010
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.8. Setting the IPv4 address

```
ptp8@ptp8-OptiPlex-5050:~$ sudo sfp_ports_config ipv6_addr set p0 FD00:0000:0000:0000:0000:0000:0000:00AA
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$ sudo sfp_ports_config ipv6_addr get p0
FD00:0000:0000:0000:0000:0000:0000:00AA
```

Figure 3.9. Setting the IPv6 address

8. For running any of the ITU-TG.8275.1 profiles, we must set the network transmission to Ethernet (L2 layer). To set the L2 layer, use the command shown in Figure 3.10.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo ptp_hw_config msg_trans set l2
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$ sudo ptp_hw_config msg_trans get
0, L2
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.10. Setting the Network Transmission Layer to L2 (Ethernet)

9. To set the clock behavior to one step or two steps, use the command shown in Figure 3.11. It should be same as in the configuration file (in this case, set the clock to one step clock.).

```
ptp8@ptp8-OptiPlex-5050:~$ sudo ptp_hw_config clk_bhr set 1step
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$ sudo ptp_hw_config clk_bhr set 2step
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.11. Setting the clock behavior to onestep or twostep

10. For the Telecom Boundary Clock in the ITU-TG.8275.1 profile, the TC type should be set to not be TC, the command is shown in Figure 3.12.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo ptp_hw_config tc_type set not
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.12. Setting the TC type to NOT as TC

11. For running the Telecom Boundary Clock, we must set the corresponding profile. To do so, we must run the following command as shown in Figure 3.13.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo engine_config profile set /home/ptp8/Desktop/TBC_8275_1.cfg
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.13. Setting the TBC profile

12. Once the profile and all the other parameters are configured, the PTP engine can be started by using the following command shown in Figure 3.14.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo engine_config status set run
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.14. Starting the PTP-Engine

13. To monitor the data, we can use multiple commands.
    a. To check only the servo data.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo servo_config all get
S,0,0,0
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.15. Monitoring the Servo Data

b. To check short snippet of the PTP-Engine.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo servo_config servo_continuous get
Debug, 1
Time,  6262 716145480
parentPortIdentity, 000000.0000.000000-0
grandMasterIdentity, 000000.0000.000000
Stats, 00, 0000000000
Stats, 01, 0000000000
Stats, 02, 0000002209
Stats, 03, 0000002213
Stats, 04, 0000000000
Stats, 05, 0000000000
Stats, 06, 0000000001
Stats, 07, 0000000000
Servo, S, 0,0,0
Status, 0, Stopped
holdover status, 0, engine is not in holdover
feedback clock, 1, is available.
gnss clock, 0, is lost.
syncE clock, 0, is lost.
ocxo clock, 1, is available.
error code, 0
GNSS status, 0, is stopped.
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.16. Short snippet of the PTP-Engine

14. Generally, the telecom boundary clock is run along with syncE.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo dpll_config clk_src set syncE0_ocxo
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$ sudo dpll_config clk_src get
0, syncE from Port 0 then OCXO
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.17. Setting the SyncE clock

15. To stop the engine, use the command shown in Figure 3.18.

```
ptp8@ptp8-OptiPlex-5050:~$ sudo engine_config status set stop
1, Execution done successfully.
ptp8@ptp8-OptiPlex-5050:~$
```

Figure 3.18. To stop the PTP-Engine

Appendix A. FPGA Register Map

A register space is provided by FPGA to interact with FPGA for configuration and data transfer. These registers are mapped to the RISC-V address space at offset 0x20000000. Table A.1, Table A.2 and Table A.3 shows the available registers from the FPGA and their descriptions.

Table A.1. Register Map FPGA

S/N	Function	Offset	Description
1	Register Interface	0x00000	General Register Interface
2	Tx Parser Data for Port 0	0x10000	Tx Packet from Port0
3	Tx Parser Data for Port 1	0x20000	Tx Packet from Port1
4	Rx Parser Data for Port 0	0x30000	Rx Packet from Port0
5	Rx Parser Data for Port 1	0x40000	Rx Packet from Port1
6	Tx Timestamp Buffer for Port 0	0x50000	Timestamp for packet sent from Port0
7	Tx Timestamp Buffer for Port 1	0x60000	Timestamp for packet sent from Port1
8	Rx Timestamp Buffer for Port 0	0x70000	Timestamp for packet received from Port0
9	Rx Timestamp Buffer for Port 1	0x80000	Timestamp for packet received from Port1
10	Tx data to GPS chip	0x90000	Tx data to GPS chip
11	Rx data from GPS	0xA0000	Rx data to GPS chip
12	Rx from PPS	0xB0000	PPS timestamp from GPS chip

Table A.2. General Purpose Register Map

S/N	Offset	Bits	Mode	Reset	Description	Interpretation
Device Information						
1	0x00000	31:0	RW	0	Test Register	
2	0x00004	31:0	RO	_	FPGA logic version number	
Port Statistics Information						
3	0x00008	31:0	RO	0	Count of PTP packet received on Port0	
4	0x0000C	31:0	RO	0	Count of PTP packet received on Port1	
5	0x00010	31:0	RO	0	Count of PTP packet transmitted from port0	
6	0x00014	31:0	RO	0	Count of PTP packet transmitted from port1	
7	0x00018	31:0	RO	0	Count of non PTP packet received on Port0	
8	0x0001C	31:0	RO	0	Count of non PTP packet received on Port1	
9	0x00020	31:0	RO	0	Count of non PTP packet transmitted from port0	
10	0x00024	31:0	RO	0	Count of non PTP packet transmitted from port1	
Delay Compensation						
11	0x00040	31:0	RW	0	Egress delay compensation	
12	0x00044	31:0	RW	0	Ingress delay compensation	
13	0x0004C	31:0	RW	0	PPS delay compensation	

S/N	Offset	Bits	Mode	Reset	Description	Interpretation
User Configuration						
14	0x0005C	2:0	RW	0	DPLL clock source and priority selection.	Refer Table A.3
15	0x00060	0	RW	0	PTP clock behaviour	0b0: One Step 0b1: Two Step
16		2:1	RW	0	PTP Message transmission type	0b00: PTP over Ethernet, 0b01: PTP over IPv4, 0b10: PTP over Ipv6
17		3	RW	0	VLAN enable	0b0: Disable 0b1: Enable
18		5:4	RW	0	Transparent Clock Type	0b00: NOT TC 0b01: E2E TC 0b10: P2P TC 0b11: Forbidden
Mux Configurations						
19	0x00064	1:0	RW	0	Mux Selection Port 0	0b00: Data from CPU 0b01: Data from MAC 0b10: Non PTP Packets from Same Port 0b11: Non PTP Packets from other Port
20		5:4	RW	0	Mux Selection Port 1	
FIFO Status						
21	0x00078	31:0	RO	0	No of PTP Packets in Rx Path 0	
22	0x0007C	31:0	RO	0	No of PTP Packets in Rx Path 1	
23	0x00080	31:0	RW	0	Length of PTP Packet -- Port 0	
24	0x00084	31:0	RW	0	Length of PTP Packet -- Port 1	
25	0x00088	0	RO	0	PTP Tx Status -- Port 0	
26	0x0008C	0	RO	0	PTP Tx Status -- Port 1	
27	0x00090	31:0	RO	0	Length of GNSS Packet	
28	0x00094	31:0	RO	0	No Of Bytes in GNSS FIFO	
29	0x00098	31:0	RO	0	No of Timestamps in the PPS FIFO	
ToD Control						
30	0x0009C	31:0	RW	0	Set seconds value low (31:0)	
31	0x000A0	15:0	RW	0	Set seconds value high (47:32)	
32	0x000A4	0	RW	0	Trigger1 to set the time	
33	0x000A8	31:0	RO	0	Get seconds value low (31:0)	
34	0x000AC	15:0	RO	0	Get seconds value high (47:32)	
35	0x000B0	31:0	RO	0	Get nanoseconds value	
36	0x000B4	0	RW	0	Trigger to get the ToD value	
37	0x000B8	31:1	RW	0	Set nanoseconds(31:0)	
DSPLL Frequency Offset						
38	0x000C0	0	RW	0	DSPLL FINC	
39	0x000C4	0	RW	0	DSPLL FDEC	
40	0x00134	9:0	RW	0	Integer Step Size	
41	0x00138	31:0	RW	0	Fraction Step Size	
IP and MAC Addresses						
42	0x000D8	31:0	RW	0	mac_addr_p0_l (31:0)	
43	0x000DC	15:0	RW	0	mac_addr_p0_h (47:32)	

S/N	Offset	Bits	Mode	Reset	Description	Interpretation
44	0x000E0	31:0	RW	0	mac_addr_p1_l (31:0)	
45	0x000E4	15:0	RW	0	mac_addr_p1_h (47:32)	
46	0x000E8	31:0	RW	0	ip_addr_p0	
47	0x000EC	31:0	RW	0	ip_addr_p1	
48	0x000F0	31:0	RW	0	ipv6_addr_p0_0 (31:0)	
49	0x000F4	31:0	RW	0	ipv6_addr_p0_1 (63:32)	
50	0x000F8	31:0	RW	0	ipv6_addr_p0_2 (95:64)	
51	0x000FC	31:0	RW	0	ipv6_addr_p0_3 (127:96)	
52	0x00100	31:0	RW	0	ipv6_addr_p1_0 (31:0)	
53	0X00104	31:0	RW	0	ipv6_addr_p1_1 (63:32)	
54	0x00108	31:0	RW	0	ipv6_addr_p1_2 (95:64)	
55	0x0010C	31:0	RW	0	ipv6_addr_p1_3 (127:96)	

Note:

1. Send a rising edge using low->high->low to trigger.

Table A.3. DPLL Clock Source Priority

S/N	Bits	Priority (Top to Bottom)
1	0b000	GNSS SyncEP0 SyncEP1 OCXO
2	0b010	GNSS OCXO
3	0b011	SyncEP0 OCXO
4	0b100	SyncEP1 OCXO
5	0b101	OCXO
6	Other	Reserved

Appendix B. RISC-V UART Frames

The middleware sends and receives frames from the UART connected on RISC-V processor. Table B.1 shows the frame format for the communication over UART. The start of frame for all the frames is set at a constant value of 0xAA.

Table B.1. UART Frame Format

S/N	Field Name	Offset	Size (Byte)	Description
1	Start of Frame (SOF)	0	1	Start of Frame Delimiter
2	Function code (FC)	1	1	Function Code
3	Data/Information	2	256	Information or DATA
4	CRC	258	1	CRC-8

For every function to be execute on the hardware there is a function code associated with it. Table B.2 shows list of the function codes recognized by the hardware.

Table B.2. Function Codes

Enum Identifier	Function code	Description
fun_debug	0xFE	Debug Frame.
fun_ver_no	0xA0	Version Number.
fun_raw_rd	0xA1	Read FPGA Register.
fun_raw_wr	0xA2	Write FPGA Register.
fun_msgq_rd	0xA3	Read Message queue.
fun_port_st	0xA4	Read the packet counter from ports.
fun_port_addr_get	0xA5	Get port addresses.
fun_port_addr_set	0xA6	Set port addresses.
fun_hw_cfg_get	0xA7	Get hardware configurations.
fun_hw_cfg_set	0xA8	Set hardware configurations.
fun_hw_time_get	0xA9	Get the hardware time.
fun_hw_time_set	0xAB	Set the hardware time.
fun_dpll_freq_get	0xAC	Get the frequency offset.
fun_dpll_freq_set	0xAD	Set the frequency offset.
fun_vlan_get	0xAE	Get the VLAN configurations.
fun_vlan_set	0xAF	Set the VLAN configurations.
fun_gnss_get	0xB0	Get the GNSS status.
fun_gnss_set	0xB1	Set the GNSS status.
fun_ds_par_get	0xB2	Get the parent dataSet.
fun_ds_port_get	0xB3	Get the port dataSet.
fun_ds_dflt_get	0xB4	Get the default dataSet.
fun_delay_get	0xB5	Get the delay compensation from engine.
fun_delay_set	0xB6	Set the delay compensation from engine.
fun_engine_get	0xB7	Get the engine status.
fun_engine_set	0xB8	Send the Engine command.
fun_servo_get	0xB9	Get the servo data.
fun_randm_frm	0x00	Random Frame for Testing.
fun_engine_hw_conf_set	0x55	Engine hardware config set.

Enum Identifier	Function code	Description
fun_engine_hw_conf_get	0x56	Engine hardware config get.
fun_arp_addr_set	0x57	Engine hardware config set.
fun_arp_addr_get	0x58	Engine hardware config get.
fun_servo_continous_get	0x59	Servo Continuous data get.
fun_phc_adj_time	0x60	PHC adjust time function code.
fun_clock_loss	0x61	DsPLL Clock loss status.
fun_holdover	0x62	PTP Engine Holdover Condition.
fun_errReport	0x63	Hardware error reporting.
fun_reset	0x64	Reset the hardware.
fun_invalid	0xFF	Invalid frame received when hardware error.

By sending the specific frame to the hardware, the hardware responds with information. These commands and responses are listed in the following tables.

Table B.3. Test Frame Command and Response

Offset	Size (Bytes)	Value Command	Description	Value Response	Response
0	1	0xAA	Start of frame delimiter	0xAA	Start of frame delimiter
1	1	0xFE	Function code for debug	0xFE	Function code for debug
2	256	0xXX	Random Test data	0xYY	1's Compliment of data
258	0	0x00	Trailing Zeros	0x00	Trailing Zeros
258	1	0xXX	CRC of frame	0xXXss	CRC of frame

Table B.4. Get Version Number

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xA0	0xA0	Version number for software and FPGA
2	1	0x00	0xXX	FW major version number
3	1	0x00	0xXX	FW minor version number
4	2	0x00	0xXX	Patch Number for Firmware
6	1	0x00	0xXX	FPGA major number
7	1	0x00	0xXX	FPGA minor number
8	2	0x00	0xXX	Patch number for FPGA bit file
10	248	0x00	0x00	Zeros
258	1	0xXXss	0xXX	CRC of frame

Table B.5. Raw Register Read

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xA1	0xA1	Raw register read
2	4	0xXXXXXXXX	0xXXXXXXXX	32 bit register address

Offset	Size	Command	Response	Description
6	4	0x00	0xYYYYYYYY	32 bit register value
10	248	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of frame

Table B.6. Raw Register Write

Offset	Size	Command	Response	Descriptions
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xA2	0xA2	Raw register write
2	4	0xXXXXXXXX	0xXXXXXXXX	32 bit register address
6	4	0xXXXXXXXX	0xXXXXXXXX	32bit register value which has been set to the register
10	248	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of frame

Table B.7. Get Port Stats

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xA4	0xA4	Read port stats
2	4	0x00	0xXXXXXXXX	Number of PTP packets received at Port0
6	4	0x00	0xXXXXXXXX	Number of PTP packets received at Port1
10	4	0x00	0xXXXXXXXX	Number of PTP packets transmitted on Port0
14	4	0x00	0xXXXXXXXX	Number of PTP packets transmitted on Port1
18	4	0x00	0xXXXXXXXX	Number of Non PTP packets received at Port0
22	4	0x00	0xXXXXXXXX	Number of Non PTP packets received at Port1
26	4	0x00	0xXXXXXXXX	Number of Non PTP packets transmitted from Port0
30	4	0x00	0xXXXXXXXX	Number of Non PTP packets transmitted from Port1
34	224	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of frame

Table B.8. Get Port Addresses

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xA5	0xA5	Read port addresses
2	6	0x00	0xXXXXXXXXXXXX	6 bytes MAC of port0
8	6	0x00	0xXXXXXXXXXXXX	6 bytes MAC of port1
14	4	0x00	0xXXXXXXXX	4 bytes of IP address for p0
18	4	0x00	0xXXXXXXXX	4 bytes of IP address for p1
22	16	0x00	0xXXXXXXXXXXXX XXXXXXXXXXXXXX XXXXXX	16 bytes of IPv6 address for p0

Offset	Size	Command	Response	Description
38	16	0x00	0xXXXXXXXXXXXX XXXXXXXXXXXXXX XXXXXX	16 bytes of IPv6 address for p1
54	204	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of frame

Table B.9. Set Port Addresses

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xA6	0xA6	Write port addresses
2	6	0xXXXXXXXXXXXX	0xXXXXXXXXXXXX	6 bytes MAC of port0
8	6	0xXXXXXXXXXXXX	0xXXXXXXXXXXXX	6 bytes MAC of port1
14	4	0xXXXXXXXX	0xXXXXXXXX	4 bytes of IP address for p0
18	4	0xXXXXXXXX	0xXXXXXXXX	4 bytes of IP address for p1
22	16	<16B IPv6>	<16B IPv6>	16 bytes of IPv6 address for p0
38	16	<16B IPv6>	<16B IPv6>	16 bytes of IPv6 address for p1
54	204	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of frame

Table B.10. Hardware Configuration Get

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xA7	0xA7	Get hardware configs
2	1	0x00	0xXX	DPLL clock source (Table B.12)
3	1	0x00	0xXX	Transmission Type (Table B.13)
4	1	0x00	0xXX	TC Type (Table B.14)
5	1	0x00	0xXX	Clock behavior (Table B.15)
6	1	0x00	0xXX	MUX p0 (Table B.16)
7	1	0x00	0xXX	MUX p1 (Table B.16)
8	250	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of frame

Table B.11. Hardware Configuration Set

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xA8	0xA8	Set hardware configs
2	1	0xXX	0xXX	DPLL clock source (Table B.12)
3	1	0xXX	0xXX	Transmission Type (Table B.13)
4	1	0xXX	0xXX	TC Type (Table B.14)
5	1	0xXX	0xXX	Clock behavior (Table B.15)

Offset	Size	Command	Response	Description
6	1	0xXX	0xXX	MUX p0 (Table B.16)
7	1	0xXX	0xXX	MUX p1 (Table B.16)
8	250	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of frame

Table B.12. DPLL Clock Source Values

Value	DPLL Clock Source (Priority)
0	SycnE Port0, OCXO
1	SyncE Port1, OCXO
2	GNSS, OCXO
3	OCXO
4	GNSS, SyncE Port0, OCXO
5	GNSS, SyncE Port1, OCXO

Table B.13. Message Transmission Type

Value	Message Transmission Type
0	IEEE802.3 (L2)
1	UDP over IPv4
2	UDP over IPv6
3	Transport Specific gPTP

Table B.14. TC Type

Value	TC Type
0	None (The clock is not a TC clock)
1	TC-E2E (The clock is an E2E TC)
2	TC-P2P (The clock is a P2P TC)

Table B.15. Clock Behaviour Type

Value	Clock Behaviour
0	One Step Clock
1	Two Step Clock

Table B.16. MUX Settings

Value	Multiplexer Setting
0	TX Traffic from CPU
1	TX Traffic from same port.
2	TX Traffic from other port.
3	TX Traffic from user logic

Table B.17. Get the Hardware Time

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xA9	0xA9	Get hardware time

Offset	Size	Command	Response	Description
2	8	0x00	<64b sec>	64 bits of seconds field
10	4	0x00	<32b ns>	32 bits of nanoseconds
14	244	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.18. Set the Hardware Time

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xAB	0xAB	Set hardware time
2	8	<64b sec>	<64b sec>	64 bits of seconds field
10	4	<32b ns>	<32b ns>	32 bits of nanoseconds
14	244	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.19. Get DPLL Frequency Offset

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xAC	0xAC	Get DPLL frequency offset
2	4	0x00	<32bit FO>	32 bits of DPLL offset in ppb
6	252	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.20. Set DPLL Frequency Offset

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xAD	0xAD	Set DPLL frequency offset
2	4	<32bit FO>	<32bit FO>	32 bits of DPLL offset in ppb
6	252	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.21. Get VLAN Settings

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xAE	0xAE	Get VLAN configs
2	1	0x00	0x0X	Status of VLAN (enable, disable, QinQ) port0 0: Disable 1: VLAN Tag 2: QinQ Tag

Offset	Size	Command	Response	Description
3	1	0x00	0x0X	Status of VLAN (enable, disable, QinQ) port1 0: Disable 1: VLAN Tag 2: QinQ Tag
4	2	0x00	0xXXXX	16 bits of VLAN TAG including DCP and PIE VID for VLAN at Port0
6	2	0x00	0xXXXX	16 bits of VLAN TAG including DCP and PIE VID for VLAN at Port1
8	2	0x00	0xXXXX	16 bits of VLAN TAG including DCP and PIE VID for QinQ at Port0
10	2	0x00	0xXXXX	16 bits of VLAN TAG including DCP and PIE VID for QinQ at Port1
12	246	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.22. Set VLAN Settings

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xAF	0xAF	Get VLAN configs
2	1	0x0X	0x0X	Status of VLAN (enable, disable, QinQ) port0 0: Disable 1: VLAN Tag 2: QinQ Tag
3	1	0x0X	0x0X	Status of VLAN (enable, disable, QinQ) port1 0: Disable 1: VLAN Tag 2: QinQ Tag
4	2	0xXXXX	0xXXXX	16 bits of VLAN TAG including DCP and PIE VID for VLAN at Port0
6	2	0xXXXX	0xXXXX	16 bits of VLAN TAG including DCP and PIE VID for VLAN at Port1
8	2	0xXXXX	0xXXXX	16 bits of VLAN TAG including DCP and PIE VID for QinQ at Port0
10	2	0xXXXX	0xXXXX	16 bits of VLAN TAG including DCP and PIE VID for QinQ at Port1
12	246	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.23. Get and Set the GNSS Synchronization

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xB0	0xB0	Get GNSS status
2	1	0x00	0xXX	Status of gnss 0: Stopped 1: Running
3	255	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.24. Get Parent DS

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xB2	0xB2	Get Parant Data set
2	8	0x00	<8Byetes>	Master port identity
10	2	0x00	<2Byetes>	Master port number
12	1	0x00	0xXX	Parent stats
13	1	0x00	0x00	Reserved in parentDS
14	2	0x00	<2Byetes>	observedParentOffsetScaledLogVariance
16	4	0x00	<4Byetes>	observedParentClockPhaseChangeRate
20	1	0x00	0xXX	grandmasterPriority1
21	1	0x00	0xXX	grandmasterClockQuality.clockClass
22	1	0x00	0xXX	grandmasterClockQuality.offsetScaledLogVariance
23	2	0x00	<2Byetes>	grandmasterClockQuality.grandmasterPriority2
25	1	0x00	0xXX	grandmasterPriority2
26	8	0x00	<8Byetes>	grandmasterIdentity. ClockIdentity
34	224	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.25. Get Port DS

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xB3	0xB3	Get port data set
2	8	0x00	<8Bytes>	Port identity.ClockIdentity for port0
10	2	0x00	<2Bytes>	Port identity.portNumber for port0
12	1	0x00	<1Byte>	portState for port0
13	1	0x00	<1Byte>	logMinDelayReqInterval for port0
14	8	0x00	<8Bytes>	peerMeanPathDelay for port0
22	1	0x00	<1Byte>	logAnnounceInterval for port0
23	1	0x00	<1Byte>	announceReceiptTimeout for port0
24	1	0x00	<1Byte>	logSyncInterval for port0
25	1	0x00	<1Byte>	delayMechanism for port0
26	1	0x00	<1Byte>	logMinPdelayReqInterval for port0
27	1	0x00	<1Byte>	versionNumber for port0
28	8	0x00	<8Bytes>	Port identity.ClockIdentity for port1
36	2	0x00	<2Bytes>	Port identity.portNumber for port1
38	1	0x00	<1Byte>	portState for port1
39	1	0x00	<1Byte>	logMinDelayReqInterval for port1
40	8	0x00	<8Bytes>	peerMeanPathDelay for port1
48	1	0x00	<1Byte>	logAnnounceInterval for port1

Offset	Size	Command	Response	Description
49	1	0x00	<1Byte>	announceReceiptTimeout for port1
50	1	0x00	<1Byte>	logSyncInterval for port1
51	1	0x00	<1Byte>	delayMechanism for port1
52	1	0x00	<1Byte>	logMinPdelayReqInterval for port1
53	1	0x00	<1Byte>	versionNumber for port1
54	204	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.26. Get Default DS

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xB4	0xB4	Get Default Dataset
2	1	0x00	<1Byte>	Flags
3	1	0x00	<1Byte>	reserved1
4	2	0x00	<2Bytes>	numberPorts
6	1	0x00	<1Byte>	priority1
7	1	0x00	<1Byte>	clockQuality.clockClass
8	8	0x00	<8Bytes>	clockQuality.clockAccuracy
16	2	0x00	<2Bytes>	clockQuality.offsetScaledLogVariance
18	1	0x00	<1Byte>	priority2
19	8	0x00	<8Bytes>	ClockIdentity
27	1	0x00	<1Byte>	domainNumber
28	1	0x00	<1Byte>	reserved2
29	229	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.27. Get Delay Compensation

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xB5	0xB5	Get latency compensation
2	4	0x00	<4Bytes>	port.tx_timestamp_offset
6	4	0x00	<4Bytes>	port.rx_timestamp_offset
10	4	0x00	<4Bytes>	PPS Offset
14	244	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.28. Set Delay Compensation

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xB6	0xB6	Get latency compensation
2	4	<4Bytes>	<4Bytes>	port.tx_timestamp_offset
6	4	<4Bytes>	<4Bytes>	port.rx_timestamp_offset
10	4	<4Bytes>	<4Bytes>	PPS Offset
14	240	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.29. Get PTP Engine Status

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xB7	0xB7	Get status of PTP Engine
2	1	0x00	0xXX	Status of PTP engine 0: Stopped 1: Running 2: Fault 3: Bug
3	255	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.30. PTP Engine Control

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0xB8	0xB8	PTP Engine control command
2	1	0x0X	0x0X	Start, stop
2	256	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.31. Get the Servo Status

Offset	size	command	Response	Description
0	1	0xAA	0xAA	Start of Frame (SOF)
1	1	0xB9	0xB9	Function Code (FC) to get servo status
2	1	0x00	<1Byte>	Format of output
3	8	0x00	<8Bytes>	Master offset nanosecond
11	8	0x00	<8Bytes>	Frequency offset
19	8	0x00	<8Bytes>	Path Delay
27	8	0x00	<8Bytes>	Master offset (rms) ns
35	8	0x00	<8Bytes>	Master offset MAX ns
43	8	0x00	<8Bytes>	Frequency offset (rms)

Offset	size	command	Response	Description
51	8	0x00	<8Bytes>	Frequency offset delta
59	8	0x00	<8Bytes>	Path delay (rms)
67	8	0x00	<8Bytes>	Path delay delta
75	1	0x00	<1Byte>	0 = SERVO_UNLOCKED, 1 = SERVO_JUMP, 2 = SERVO_LOCKED, 3 = SERVO_LOCKED_STABLE
76	182	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC

Table B.32. Error Response

Offset	Size	Value	Description
0	1	0xAA	Start of frame delimiter
1	1	0xFF	Error frame function code
2	4	0xDEADBEEF	When incorrect frame received data to send from RISCV
6	252	0x00	Trailing Zeros
258	1	0xXX	CRC of Frame

Table B.33. Set Engine Configuration

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0x55	0x55	Function code
2	1	0xXX	0xXX	Configuration Index (Table B.35)
3	255	<255bytes>	<255bytes>	Data for configuration
258	1	0xXX	0xXX	CRC of Frame

Table B.34. Get Engine Configuration

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0x56	0x56	Function code
2	1	0xXX	0xXX	Configuration Index (Table B.35)
3	255	0x00	<255bytes>	Data for configuration (Table B.35)
258	1	0xXX	0xXX	CRC of Frame

Table B.35. PTP Configurations

Config index	Data	Name	Type	Default	Description
0x01	1 byte	announceReceiptTimeout	INT	3	Number of missed announce messages before participating again into BMCA.
0x02	1 byte	asCapable	ENU	auto	0: AS_CAPABLE_TRUE 1: AS_CAPABLE_AUTO

Config index	Data	Name	Type	Default	Description
0x03	1 byte	assume_two_step	INT	0	0: Don't Assume Two Step 1: Assume Two Step
0x04	1 byte	boundary_clock_jbod	INT	0	0: The clock is not a Boundary clock. 1: The clock is a boundary clock.
0x05	1 byte	BMCA	ENU	ptp	0: Use PTP BMCA 1: Don't use BMCA
0x06	1 byte	check_fup_sync	INT	0	
0x07	1 byte	clientOnly	INT	0	0: Node can be a server 1: Node will always clicent.
0x08	1 byte	clockAccuracy	INT	0xFE	Refer IEEE1588-2019
0x09	1 byte	clockClass	INT	248	Refer IEEE1588-2019
0x0A	18 byte	clockIdentity	STR		Refer IEEE1588-2019
0x0B	1 byte	clock_class_threshold	INT	248	Refer IEEE1588-2019
0x0D	1 byte	clock_type	ENU	OC	0: OC 1: BC 2: P2P_TC 3: E2E_TC
0x0E	1 byte	dataset_comparision	ENU		
0x0F	4 byte	delayAsymmetry	INT	0	
0x10	1 byte	delay_filter	ENU	moving_median	
0x11	4 byte	delay_filter_length	INT	10	
0x12	1 byte	delay_mechanism	ENU	E2E	0: Auto 1: E2E 2: P2P 3: No Delay Mechanism
0x13	1 byte	delay_response_timeout	INT	0	
0x14	1 byte	dscp_event	INT	0	
0x15	1 byte	dscp_general	INT	0	
0x16	1 byte	domainNumber	INT	0	
0x17	4 byte	egressLatency	INT	0	
0x18	4 byte	fault_badpeernet_interval	INT		
0x19	1 byte	fault_reset_interval	INT	4	
0x1A	8 byte	first_step_threshold	DBL	0.00002	
0x1B	1 byte	follow_up_info	INT	0	
0x1C	1 byte	free_running	INT	0	
0x1D	4 byte	freq_est_interval	INT	1	
0x1E	1 byte	G.8275.defaultDS.localPriority	INT	128	
0x1F	1 byte	G.8275.portDS.localPriority	INT	128	
0x20	1 byte	gmCapable	INT		
0x21	1 byte	hwts_filter	ENU		

Config index	Data	Name	Type	Default	Description
0x22	1 byte	hybrid_e2e	INT	0	
0x23	1 byte	ignore_source_id	INT	0	
0x24	1 byte	ignore_transport_specific	INT		
0x25	4 byte	ingress_Latency	INT		
0x26	1 byte	inhibit_announce	INT	0	
0x27	1 byte	inhibit_delay_req	INT	0	
0x28	1 byte	inhibit_multicast_service	INT	0	
0x29	4 byte	initial_delay	INT		
0x2A	1 byte	kernel_leap	INT	1	
0x2B	1 byte	logAnnounceInterval	INT	1	
0x2C	1 byte	logMinDelayReqInterval	INT	0	
0x2D	1 byte	logMinPdelayReqInterval	INT	0	
0x2E	1 byte	logSyncInterval	INT	0	
0x2F	1 byte	masterOnly	INT		
0x2F	1 byte	logging_level	INT	6	
0x31	1 byte	masterOnly	INT		
0x32	1 byte	maxStepsRemoved	INT	255	
0x33	5 byte	message_tag	STR		
0x34	9 byte	manufacturerIdentity	STR	0:00:00	
0x35	4byte	max_frequency	INT	900000000	
0x36	4byte	min_neighbor_prop_delay	INT		
0x37	1 byte	msg_interval_request	INT	0	
0x38	4 byte	neighborPropDelayThresh	INT	20000000	
0x39	1 byte	net_sync_monitor	INT	0	
0x3A	1 byte	network_transport	ENU	UDPv4	0: UDS 1: UDPv4 2: UDPv6 3: L2
0x3B	4 byte	ntpshm_segment	INT	0	
0x3C	2 byte	offsetScaledLogVariance	INT	0xFFFF	
0x3D	1 byte	operLogPdelayReqInterval	INT	0	
0x3E	1 byte	operLogSyncInterval	INT	0	
0x3F	1 byte	path_trace_enabled	INT	0	
0x40	8 byte	pi_integral_const	DBL	0.0	
0x41	8 byte	pi_integral_exponent	DBL	0.4	
0x42	8 byte	pi_integral_norm_max	DBL	0.3	
0x43	8 byte	pi_integral_scale	DBL	0.0	
0x44	8 byte	pi_proportional_const	DBL	0.0	
0x45	8 byte	pi_proportional_exponent	DBL	-0.3	

Config index	Data	Name	Type	Default	Description
0x46	8 byte	pi_proportional_norm_max	DBL	0.7	
0x47	8 byte	pi_proportional_scale	DBL	0.0	
0x48	1 byte	priority1	INT	128	
0x49	1 byte	priority2	INT	128	
0x4A	10 byte	productDescription	STR	;;	
0x4B	18 byte	ptp_dst_mac	STR	01:1B:19:00:00:00	
0x4C	18 byte	p2p_dst_mac	STR	01:80:C2:00:00:0E	
0x4D	10 byte	revisionData	STR	;;	
0x4E	4 byte	sanity_freq_limit	INT	200000000	
0x4F	1 byte	serverOnly	INT	0	
0x50	4 byte	servo_num_offset_values	INT	10	
0x51	4 byte	servo_offset_threshold	INT	0	
0x52	10 byte	slave_event_monitor	STR		
0x53	1 byte	slaveOnly	INT		
0x54	1 byte	socket_priority	INT	0	
0x55	8 byte	step_threshold	DBL	0.0	
0x56	4 byte	step_window	INT		
0x57	4 byte	summary_interval	INT	0	
0x58	1 byte	syncReceiptTimeout	INT	0	
0x59	1 byte	tc_spanning_tree	INT	0	
0x5A	1 byte	timeSource	INT	0xA0	
0x5B	1 byte	time_stamping	ENU	hardware	
0x5C	1 byte	transportSpecific	INT	0x0	
0x5D	4 byte	ts2phc.channel	INT		
0x5E	4 byte	ts2phc.extts_correction	INT		
0x5F	1 byte	ts2phc.extts_polarity	ENU		
0x60	1 byte	ts2phc.master	INT		
0x61	4 byte	ts2phc.nmea_baudrate	INT		
0x62	10 byte	ts2phc.nmea_remote_host	STR		
0x63	10 byte	ts2phc.nmea_remote_port	STR		
0x64	12 byte	ts2phc.nmea_serialport	STR		
0x65	4 byte	ts2phc.pin_index	INT		
0x66	4 byte	ts2phc.pulsewidth	INT		
0x67	1 byte	tsproc_mode	ENU		
0x68	1 byte	twoStepFlag	INT	1	
0x69	4 byte	tx_timestamp_timeout	INT	10	
0x6A	1 byte	udp_ttl	INT	1	
0x6B	1 byte	udp6_scope	INT	0x0E	

Config index	Data	Name	Type	Default	Description
0x6E	1 byte	unicast_listen	INT	0	
0x6F	4 byte	unicast_master_table	INT	0	
0x70	4 byte	unicast_req_duration	INT	3600	
0x71	1 byte	use_syslog	INT	1	
0x72	10 byte	userDescription	STR	;	
0x73	4 byte	utc_offset	INT	37	
0x74	1 byte	verbose	INT	0	
0x75	1 byte	write_phase_mode	INT	0	
0x76	35 byte	unicast master table data	INT		
0x77	1 byte	set port0	INT		
0x78	1 byte	set port1	INT		

Table B.36. Get ARP Entry

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0x58	0x58	Function code
2	1	<1Byte>	<1Byte>	1 bytes arp addr index. (0 to 9, 10 ARP Indices are available)
3	6	0x00	<6Bytes>	6 bytes mac address
9	4	0x00	<4Bytes>	4 bytes ipv4 address
13	16	0x00	<16Bytes>	16 bytes ipv6 address
29	229	0x00	0x00	Trailing Zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.37. Set ARP Entry

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0x58	0x57	Function code
2	1	<1Byte>	<1Byte>	1 bytes arp addr index. (0 to 9, 10 ARP Indices are available)
3	6	<6Bytes>	<6Bytes>	6 bytes mac address
9	4	<4Bytes>	<4Bytes>	4 bytes ipv4 address
13	16	<16Bytes>	<16Bytes>	16 bytes ipv6 address
29	229	0x00	0x00	Trailing Zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.38. Servo Continuous Data

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of Frame(SOF)
1	1	0x59	0x59	Function Code(FC)

Offset	Size	Command	Response	Description
2	12	0x00	<12Bytes>	TimeStampFrameData
14	27	0x00	<27Bytes>	DefaultDSFrameData
41	32	0x00	<32Bytes>	ParentDSFrameData
73	52	0x00	<52Bytes>	PortStatsFrameData
125	85	0x00	<85Bytes>	ServoFrameData
210	13	0x00	<13Bytes>	EnginedelayFrameData
223	1	0x00	<1Byte>	ServoState
224	1	0x00	<1Byte>	Clock loss status
225	1	0x00	<1Byte>	Holdover status
226	1	0x00	<1Byte>	Error status
227	2	0x00	<2Byte>	gnssStatusFrameData
229	29	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC

Table B.39. Set Unicast Table

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0x55	0x55	Config set function code
2	1	0x76	0x76	Index value to set unicast table
3	4	<4Bytes>	<4Bytes>	Table id of unicast table
7	4	<4Bytes>	<4Bytes>	logqueryinterval data
11	4	<4Bytes>	<4Bytes>	Transmit type data (UDPv4 , UDPv6)
15	4	<4Bytes>	<4Bytes>	ipv4 address
19	4	<4Bytes>	<4Bytes>	ipv4 address
23	4	<4Bytes>	<4Bytes>	ipv4 address
27	16	<16Bytes>	<16Bytes>	ipv6 address
43	16	<16Bytes>	<16Bytes>	ipv6 address
59	16	<16Bytes>	<16Bytes>	ipv6 address
75	183	0x00	0x00	Trailing Zeros
258	1	0xXX	0xXX	CRC

Table B.40. Adjust Hardware Time

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0x60	0x60	Adjust hardware time function code
2	8	<8Bytes>	<8Bytes>	64 bits of nanoseconds to be set(tmv)
10	248	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.41. Get DPLL Clock Loss Status

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0x61	0x61	Get clock loss status function code
2	1	0x00	0xXX	Clock loss status
3	255	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.42. Get Holdover State

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0x62	0x62	Get holdover status function code
2	1	0x00	<1Byte>	Holdover Status
3	255	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Table B.43. Reset Hardware

Offset	Size	Command	Response	Description
0	1	0xAA	0xAA	Start of frame delimiter
1	1	0x63	0x63	Get clock loss status function code
2	1	0x00	<1Byte>	Reset the Hardware
3	255	0x00	0x00	Trailing zeros
258	1	0xXX	0xXX	CRC of Frame

Appendix C. Middleware Commands

The middleware is a collection of programs that are used to communicate with the RISC-V over UART. Table C.1 shows the commands, and their description that can be used to communicate.

Table C.1. Middleware Commands

S/N	Category	Commands
1	Daemon Configuration	sudo demon_config device_list get
2		sudo demon_config device_select set <index>
3	General Configurations	sudo general_config debug get
4		sudo general_config hw_ver_no get
5		sudo general_config fw_ver_no get
6		sudo general_config error get
7		sudo general_config raw_register get <register_addr>
8		sudo general_config raw_register set <register_addr> <value>
9	SFP Port Configurations	sudo sfp_ports_config port_stats get
10		sudo sfp_ports_config mac_addr get p0
11		sudo sfp_ports_config ipv4_addr get p0
12		sudo sfp_ports_config ipv6_addr get p0
13		sudo sfp_ports_config mac_addr set p0 <mac_addr>
14		sudo sfp_ports_config ipv4_addr set p0 <ipv4_addr>
15		sudo sfp_ports_config ipv6_addr set p0 <ipv6_addr>
16		sudo sfp_ports_config mac_addr get p1
17		sudo sfp_ports_config ipv4_addr get p1
18		sudo sfp_ports_config ipv6_addr get p1
19		sudo sfp_ports_config mac_addr set p1 <mac_addr>
20		sudo sfp_ports_config ipv4_addr set p1 <ipv4_addr>
21		sudo sfp_ports_config ipv6_addr set p1 <ipv6_addr>
22	DPLL Configurations	sudo dpll_config clk_src get
23		sudo dpll_config clk_src set ocxo
24		sudo dpll_config clk_src set syncE0_ocxo
25		sudo dpll_config clk_src set syncE1_ocxo
26		sudo dpll_config clk_src set gnss_ocxo
27		sudo dpll_config clk_src set gnss_syncE0_ocxo
28		sudo dpll_config clk_src set gnss_syncE1_ocxo
29		sudo dpll_config freq_offset get

S/N	Category	Commands
30		sudo dpll_config freq_offset set <frequency>
31		sudo dpll_config clock_loss get
32		sudo dpll_config lock_loss get
33	PTP HW Configurations	sudo ptp_hw_config msg_trans get
34		sudo ptp_hw_config msg_trans set l2
35		sudo ptp_hw_config msg_trans set udpv4
36		sudo ptp_hw_config msg_trans set udpv6
37		sudo ptp_hw_config msg_trans set gPTP
38		sudo ptp_hw_config tc_type get
39		sudo ptp_hw_config tc_type set not
40		sudo ptp_hw_config tc_type set e2e
40		sudo ptp_hw_config tc_type set p2p
42		sudo ptp_hw_config clk_bhr get
43		sudo ptp_hw_config clk_bhr set 1step
44		sudo ptp_hw_config clk_bhr set 2step
45	Multiplexer Configuration	sudo mux_config port0 get
46		sudo mux_config port0 set data_cpu
47		sudo mux_config port0 set non_ptp_p0
48		sudo mux_config port0 set non_ptp_p1
49		sudo mux_config port0 set user_traffic
50		sudo mux_config port1 get
51		sudo mux_config port1 set data_cpu
52		sudo mux_config port1 set non_ptp_p0
53		sudo mux_config port1 set non_ptp_p1
54		sudo mux_config port1 set user_traffic
55	VLAN Configurations	sudo vlan_config port0 get
56		sudo vlan_config port0 set disable
57		sudo vlan_config port0 set vlan <value>
58		sudo vlan_config port0 set qinq <value> <value>
59		sudo vlan_config port1 get
60		sudo vlan_config port1 set disable

S/N	Category	Commands
61		sudo vlan_config port1 set vlan <value>
62		sudo vlan_config port1 set qinq <value> <value>
63	GNSS Configurations	sudo gnss_config status get
64		sudo gnss_config status set restart
65		sudo gnss_config status set stop
66	PTP Engine Configurations	sudo engine_config status get
67		sudo engine_config profile set <file_path>
68		sudo engine_config status set run
69		sudo engine_config status set stop
70		sudo engine_config ds get parent
71		sudo engine_config ds get p0
72		sudo engine_config ds get p1
73		sudo engine_config ds get default
74		sudo engine_config delays get
75		sudo engine_config holdover get
76	PTP Servo Configurations	sudo servo_config format_type get
77		sudo servo_config master_offset get
78		sudo servo_config rms_offset get
79		sudo servo_config max_offset get
80		sudo servo_config freq_offset get
81		sudo servo_config rms_freq_offset get
82		sudo servo_config delta_freq get
83		sudo servo_config path_delay get
84		sudo servo_config rms_delay get
85		sudo servo_config delta_delay get
86		sudo servo_config all get
87		sudo servo_config servo_continuous get
88	ARP Configurations	sudo arp_addr_config set <index> <macaddr> <ipv4addr> <ipv6addr>
89		sudo arp_addr_config get <index>
90	PTP HW Time Configurations	sudo hw_time_config hw get
91		sudo hw_time_config hw set <value>
92		sudo hw_time_config hw set

S/N	Category	Commands
93		sudo hw_time_config adj set <adj_value_ns>
94	Delay Configurations	sudo delay_config egress get
95		sudo delay_config ingress get
96		sudo delay_config pps get
97		sudo delay_config egress set <value>
98		sudo delay_config ingress set <value>
99		sudo delay_config pps set <value>
100		sudo delay_config all set <egress> <ingress> <pps>
101		sudo delay_config all get

References

For more information refer to:

- Lattice ORAN™ Solution Stack web page
- Lattice Insights for Lattice Semiconductor training courses and learning plans

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

For frequently asked questions, refer to the Lattice Answer Database at www.latticesemi.com/Support/AnswerDatabase.

Revision History

Revision 1.0, January 2024

Section	Change Summary
All	Initial release.

www.latticesemi.com