*Application Note*

# *LMK5XXXXXS1 Network Synchronizer Compliance Test Report for PTP Profiles G.8275.1 and G.8275.2*

### ABSTRACT

This application note describes the performance tests and measurement results to verify the compliance of the TI network synchronizer devices (LMK5CXXXXXA, LMK5BXXXXX, and LMK5XXXXXXS1) in combination with the Oregano Systems syn1588® technology Precision Time Protocol (PTP) telecommunication profiles G.8275.1 and G.8275.2, respectively. The G.8275.1 profile covers networks with full timing support, in which all network devices provide either the PTP Boundary Clock or PTP Transparent Clock support with sufficient accuracy. Meanwhile, the G.8275.2 profile defines the PTP for networks with partial timing support, in which only some of the network devices require the PTP support. The LMK5xxxxxS1 is used for adjusting the frequency of the PTP hardware Time-of-Day (ToD) clock and is in compliance with the telecommunication accuracies of Class A (100ns), Class B (70ns), Class C (30ns), and even Class D.

## Table of Contents

## Trademarks

syn1588® is a registered trademark of Oregano Systems.
Intel® and Arria® are registered trademarks of Intel.
All trademarks are the property of their respective owners.

# 1 Hardware Architecture

The software is setup using the syn1588® PTP technology, which is ported onto an Intel® Arria® 10 SoC FPGA (10AS066K3F40E2SG). The commercially available HAN Pilot Platform, offered by terasIC, is selected to minimize the overall design effort for this project. The block diagram of the complete FPGA and clocking for a single 10G Ethernet port is shown in Figure 1-1. For the 10G Ethernet interface port, the respective hard IP cores (PMA, PCS) are configured accordingly. A MAC IP core developed by Oregano Systems is connected to the 32-bit wide XGMII interface. The PTP IP core contains the PTP ToD clock along with a set of packet scanning engines that search for PTP event messages. To account for different network communication protocols (Layer 2, IPv4, IPv6 VLAN, and so forth), the scanning engines are user-configurable with the respective pattern and mask RAM blocks. All units and modules are connected through an AXI bus interface to the embedded ARM CPU.

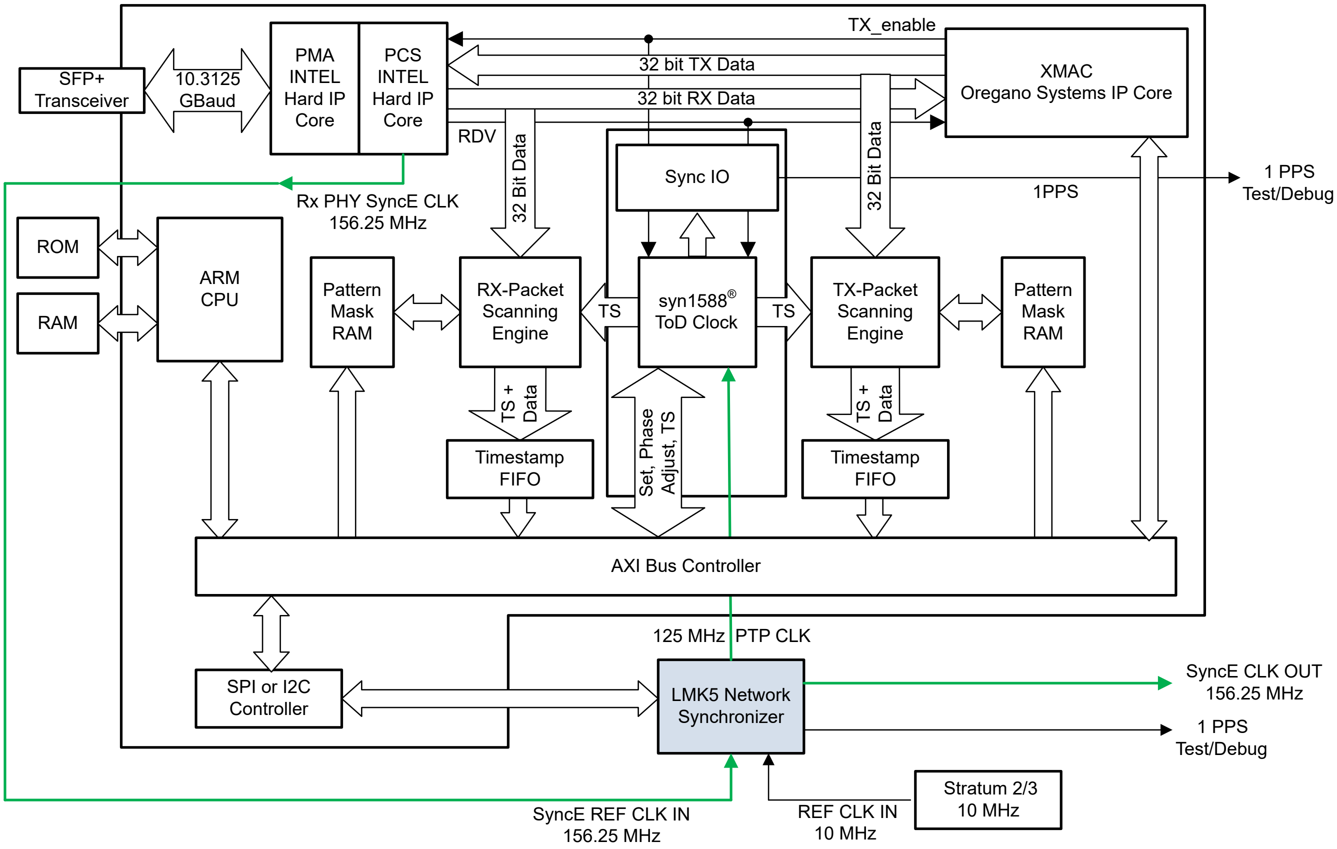


**Figure 1-1. System Block Diagram of Arria® 10 FPGA and Clocking With the syn1588® IP Core and MAC**

After porting the Oregano syn1588® technology onto the Arria® 10 FPGA, both the hardware and software of the syn1588® technology are enhanced to use the digitally tunable network synchronizer (LMK5XXXXXS1). A standard SPI port provided by the Arria® 10 SoC FPGA is used to establish a bidirectional communication with the LMK5XXXXXS1 for configuration, status monitoring, and phase and frequency tuning through the digitally-controlled oscillator (DCO).

## 1.1 Clocking Scheme

The LMK5XXXXXS1 is configured to provide differential 125MHz network timing PTP clock to the syn1588® Time-of-Day clock in the LVDS output format. The 125MHz frequency and phase is adjusted digitally using the built-in DCO function of the digital phase-locked loop (DPLL) in the network synchronizer. The reference input frequencies of the DPLL are configured as 10MHz and 156.25MHz. The 10MHz signal is externally generated by an OCXO or a Rubidium stable lab reference clock and takes priority over the 156.25MHz signal, which is a PHY recovered clock frequency that is derived from the 10G Ethernet transceivers.

The 156.25MHz recovered PHY clock signal is extracted from the PCS modules and supplied to the DPLL reference input of the LMK5XXXXXS1 in LVDS format. If the PTP device, which provides the time information, can provide a sufficiently stable carrier frequency for SyncE, then the 156.25MHz clock is selected as the primary reference for the PTP DPLL within the LMK5XXXXXS1.

Figure 1-2 provides a block diagram of the LMK5XXXXXS1 configuration. The digital frequency adjustment, as well as the input and output clock configuration, of the network synchronizer can be tested using the corresponding TICS Pro profile of the LMK5XXXXXS1. The register configuration of the LMK5XXXXXS1 can be visualized using the TICS Pro software as depicted in Figure 1-4.

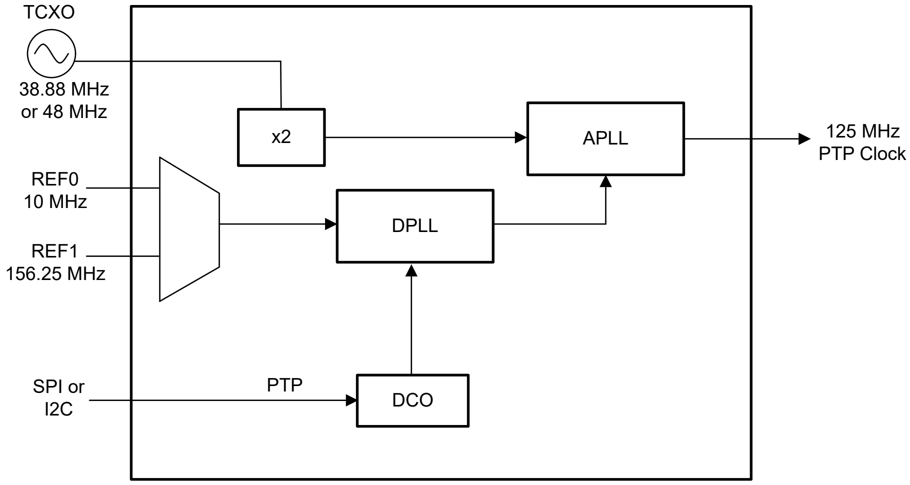


**Figure 1-2. LMK5XXXXXS1 Input and Output Clock Configuration Overview**

The DCO is enabled for the DPLL loop to phase and frequency steer the output clock, as required for IEEE-1588 PTP. The desired frequency step size has a frequency accuracy of a parts-per-trillion (ppt). A more detailed overview depicting the interconnection between DPLL2 and APLL2 is shown in Figure 1-3. The TICS Pro configuration of DPLL2 and APLL2 is represented in Figure 1-4.

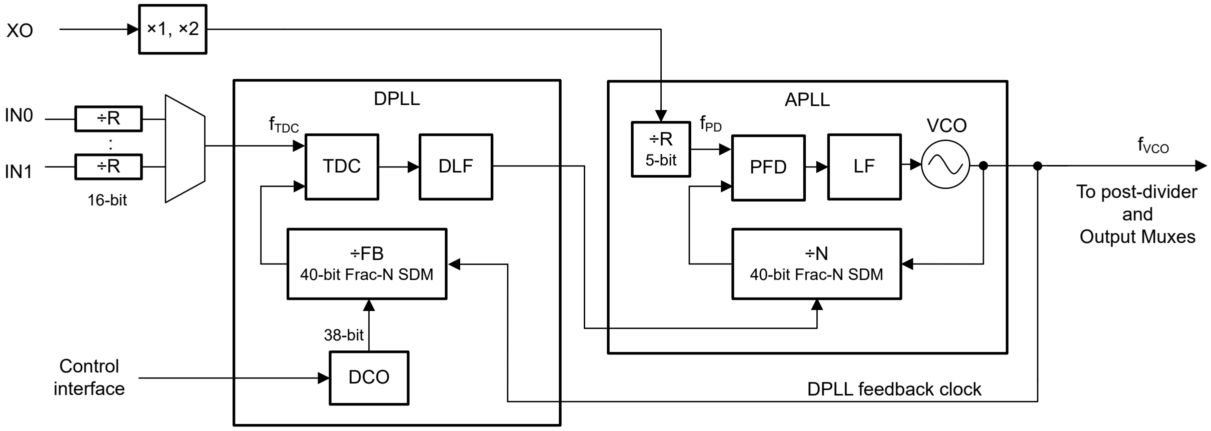
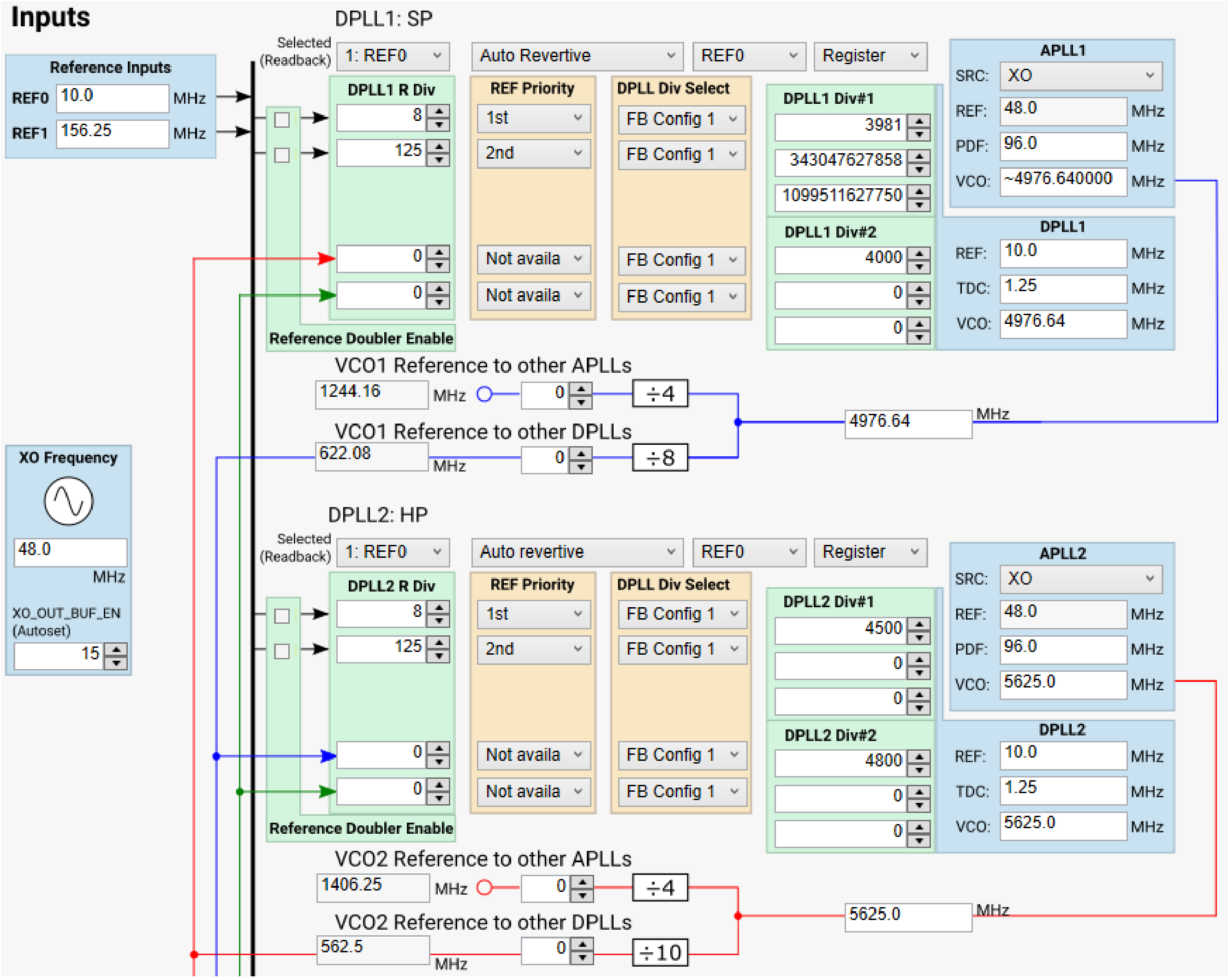


**Figure 1-3. DPLL and APLL in the LMK5XXXXXS1 Block Diagram**



Inputs

Reference Inputs
REF0 10.0 MHz
REF1 156.25 MHz

XO Frequency
48.0 MHz
XO_OUT_BUF_EN (Autoset) 15

DPLL1: SP
Selected (Readback) 1: REF0 | Auto Revertive | REF0 | Register

| DPLL1 R Div | REF Priority | DPLL Div Select |
|---|---|---|
| 8 | 1st | FB Config 1 |
| 125 | 2nd | FB Config 1 |
| 0 | Not availa | FB Config 1 |
| 0 | Not availa | FB Config 1 |

Reference Doubler Enable

DPLL1 Div#1: 3981, 343047627858, 1099511627750
DPLL1 Div#2: 4000, 0, 0

APLL1
SRC: XO
REF: 48.0 MHz
PDF: 96.0 MHz
VCO: ~4976.640000 MHz

DPLL1
REF: 10.0 MHz
TDC: 1.25 MHz
VCO: 4976.64 MHz

VCO1 Reference to other APLLs
1244.16 MHz ○ 0 ÷4

VCO1 Reference to other DPLLs
622.08 MHz 0 ÷8

4976.64 MHz

DPLL2: HP
Selected (Readback) 1: REF0 | Auto revertive | REF0 | Register

| DPLL2 R Div | REF Priority | DPLL Div Select |
|---|---|---|
| 8 | 1st | FB Config 1 |
| 125 | 2nd | FB Config 1 |
| 0 | Not availa | FB Config 1 |
| 0 | Not availa | FB Config 1 |

Reference Doubler Enable

DPLL2 Div#1: 4500, 0, 0
DPLL2 Div#2: 4800, 0, 0

APLL2
SRC: XO
REF: 48.0 MHz
PDF: 96.0 MHz
VCO: 5625.0 MHz

DPLL2
REF: 10.0 MHz
TDC: 1.25 MHz
VCO: 5625.0 MHz

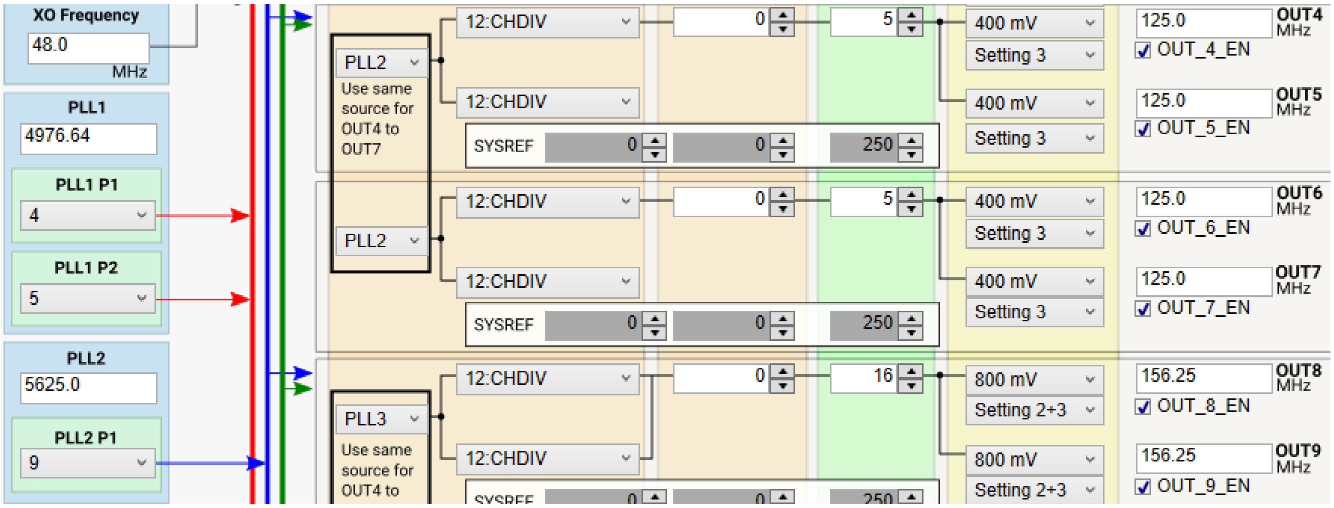
VCO2 Reference to other APLLs
1406.25 MHz ○ 0 ÷4

VCO2 Reference to other DPLLs
562.5 MHz 0 ÷10

5625.0 MHz

**Figure 1-4. DPLL Configuration**

For compliance testing purposes, only one differential output port, OUT7, is enabled in LVDS mode as shown in Figure 1-5. The network synchronizer device parameters are loaded into the control registers at startup via SPI or I2C.



XO Frequency 48.0 MHz
PLL1 4976.64
PLL1 P1 4
PLL1 P2 5
PLL2 5625.0
PLL2 P1 9

PLL2 — Use same source for OUT4 to OUT7
PLL2
PLL3 — Use same source for OUT4 to

| | | | | | |
|---|---|---|---|---|---|
| 12:CHDIV | 0 | 5 | 400 mV<br>Setting 3 | 125.0 MHz | OUT4 ☑ OUT_4_EN |
| 12:CHDIV<br>SYSREF | 0 | 250 | 400 mV<br>Setting 3 | 125.0 MHz | OUT5 ☑ OUT_5_EN |
| 12:CHDIV | 0 | 5 | 400 mV<br>Setting 3 | 125.0 MHz | OUT6 ☑ OUT_6_EN |
| 12:CHDIV<br>SYSREF | 0 | 250 | 400 mV<br>Setting 3 | 125.0 MHz | OUT7 ☑ OUT_7_EN |
| 12:CHDIV | 0 | 16 | 800 mV<br>Setting 2+3 | 156.25 MHz | OUT8 ☑ OUT_8_EN |
| 12:CHDIV<br>SYSREF | 0 | 250 | 800 mV<br>Setting 2+3 | 156.25 MHz | OUT9 ☑ OUT_9_EN |

**Figure 1-5. Frequency Output Configuration**

## 1.2 FPGA Design

The syn1588® IP core is easily ported to the FPGA device using a standard software tool chain. The timing report of the compiled design shows that the internal syn1588® ToD clock can operate at a system frequency of up to 250MHz. The 125MHz is chosen for the first implementation because this frequency is a common use case for ToD implementations. The 125MHz network timing PTP clock can be increased to 250MHz by reducing the output divider setting by 2 in the LMK5XXXXXS1 or doubling the 125MHz clock through an internal PLL in the Arria® 10 FPGA. Increasing the frequency to 250MHz improves the resolution of the PTP clock and the time stamp resolution without introducing any noticeable jitter.

The following list, taken from the FPGA fitter report, shows resource utilization of the FPGA design with two 10G Ethernet ports with full PTP support. For this implementation, a second pair of PCS PMA modules is instantiated with an XMAC IP core attached. While the second 10G Ethernet port requires an independent pair of packet scanning engines, the Ethernet port shares the syn1588® hardware ToD clock, which is extended with two additional time stamp registers.

- Fitter Status: Successful - Tue Dec 14 13:37:50 2021
- Quartus Prime Version: 21.3.0 Build 170 09/23/2021
- Family: Arria 10 Device: 10AS066K3F40E2SG
- Final Logic utilization (in ALMs): 33,223 / 251,680 (13 %)
- Total registers: 55308 Total pins: 464 / 864 (54 %)
- Total block memory bits: 5,402,112 / 43,642,880 (12 %)
- Total RAM Blocks: 364 / 2,131 (17 %)
- Total DSP Blocks: 0 / 1,687 (0 %)
- Total HSSI RX channels: 2 / 36 (6 %)
- Total HSSI TX channels: 2 / 36 (6 %)
- Total PLLs: 10 / 80 (13 %)

# 2 syn1588® Synchronization Algorithm

The syn1588® PTP Stack waits for PTP messages from a PTP Grandmaster after performing basic initialization tasks, such as initializing the network synchronizer. When the bidirectional time transfer is established, the PTP Stack calculates the offset of the local clock with respect to the leading clock. For larger initial offsets exceeding a user-definable boundary, the PTP stack adjusts the hardware ToD clock asynchronously, by directly updating the time information. The PTP stack then calculates and adjusts the frequency offset of the local oscillator using a series of SYNC messages. Subsequently, the PTP stack corrects the residual frequency offset. The final two steps of the adjustment process are done synchronously using a dedicated hardware module of the IP core. As a result, the PTP Stack is able to adjust the clock of the PTP Follower to less than 20ns prior to engaging the PTP control servo loop.

## 2.1 PTP Time-of-Day Clock Adjustment Algorithm

When the PTP Stack is initialized in *cold start mode*, all of the hardware registers (both of the network synchronizer and the PTP Time-of-Day (ToD) hardware clock) must be uninitialized. After updating the PTP hardware clock registers with the respective startup value, the complete register set of the network synchronizer is written to initialize the device. The startup register initialization values can be generated in hex-file format using the TICS Pro software.

The PTP hardware Time-of-Day clock in steady state is adjusted solely by varying the network timing PTP reference input frequency. The PTP clock servo loop modifies the output frequency by programming the internal DCO in the LMK5XXXXXS1. The corresponding example code section of the syn1588® PTP Stack software is shown below. In the following example code, two functions have been implemented: *getDrift* and *setDrift*. The *getDrift* function reads the current value of the frequency adjustment from the network synchronizer while the *setDrift* function updates the frequency adjustment with a user-provided value.

The raw input data extracted from the time stamps is reformatted and scaled to ns/s using a scale factor derived from the TICS Pro software. The time information containing the PTP event messages is propagated through prefilters and sent to the phase interpolation (PI) servo which calculates a new frequency adjustment value. The new frequency value is passed to the function setDrift, which reads the current frequency offset from the network synchronizer, calculates a new drift value, rescales and reformats the drift accordingly, and updates the DPLL with both the new DCO adjustment value and whether to increment or decrement the DCO frequency.

```
static constexpr double magic_factor = 247390116249 / 100000.0;

ptp::scalednanoseconds ptp::ti::Osc::getDrift() const
{
  auto values = readRegs(ptp::ti::Register::DPLL2_FB_NUM_STAT_4, 4);
  std::uint64_t value = 0;
  value |= values[0];
  value |= static_cast<std::uint64_t>(values[1]) << 8;
  value |= static_cast<std::uint64_t>(values[2]) << 16;
  value |= static_cast<std::uint64_t>(values[3]) << 24;

  // if MSB is 1 handle two's complement
  if (value > (1ull << 39)) value = -((value ^ 0xffffffffff) + 1);

  auto drift = std::chrono::duration<double>(value / magic_factor);
  return std::chrono::duration_cast<ptp::scalednanoseconds>(drift);
}

void ptp::ti::Osc::setDrift(ptp::scalednanoseconds drift)
{
    // Check for absolute limits
  if(drift > std::chrono::microseconds(400)) drift = std::chrono::microseconds(400);
  if(drift < std::chrono::microseconds(-400)) drift = std::chrono::microseconds(-400);

  ptp::scalednanoseconds drift_diff = drift - getDrift();
  if(drift_diff == std::chrono::seconds(0)) {return;}    // nothing to do

  // check for limit per change
  if(drift_diff > std::chrono::nanoseconds(40000)) drift_diff = std::chrono::nanoseconds(40000);
  if(drift_diff < std::chrono::nanoseconds(-40000)) drift_diff = std::chrono::nanoseconds(-40000);

  auto driff_f = std::chrono::duration_cast<std::chrono::duration<double,
std::nano>>(drift_diff).count();

  auto value = static_cast<std::uint64_t>(magic_factor * std::abs(driff_f));
  auto values = std::vector<std::uint8_t>(5, 0x00);
  std::copy(reinterpret_cast<std::uint8_t *>(&value),reinterpret_cast<std::uint8_t *>(&value)+5,
values.rbegin());

  writeRegs(ptp::ti::Register::DPLL2_FBFDEV_BY4, values);

  // if we have a positive drift, the PLL has to be slowed down 1 is decrement, 0 is increment speed
  std::uint32_t direction = drift_diff.count() > 0 ? 1: 0;
  writeReg(ptp::ti::Register::DPLL2_FBFDEVUPDATE, direction);

  m_currentDrift = m_currentDrift + drift_diff;
}
```

## 3 Test Setup

The PTP implementation on the Intel® FPGA is functionally tested by mounting the LMK5XXXXXS1 M2 module onto an interface connector board to match the FMC connector of the HAN Pilot platform. The setup is captured in Figure 3-1.

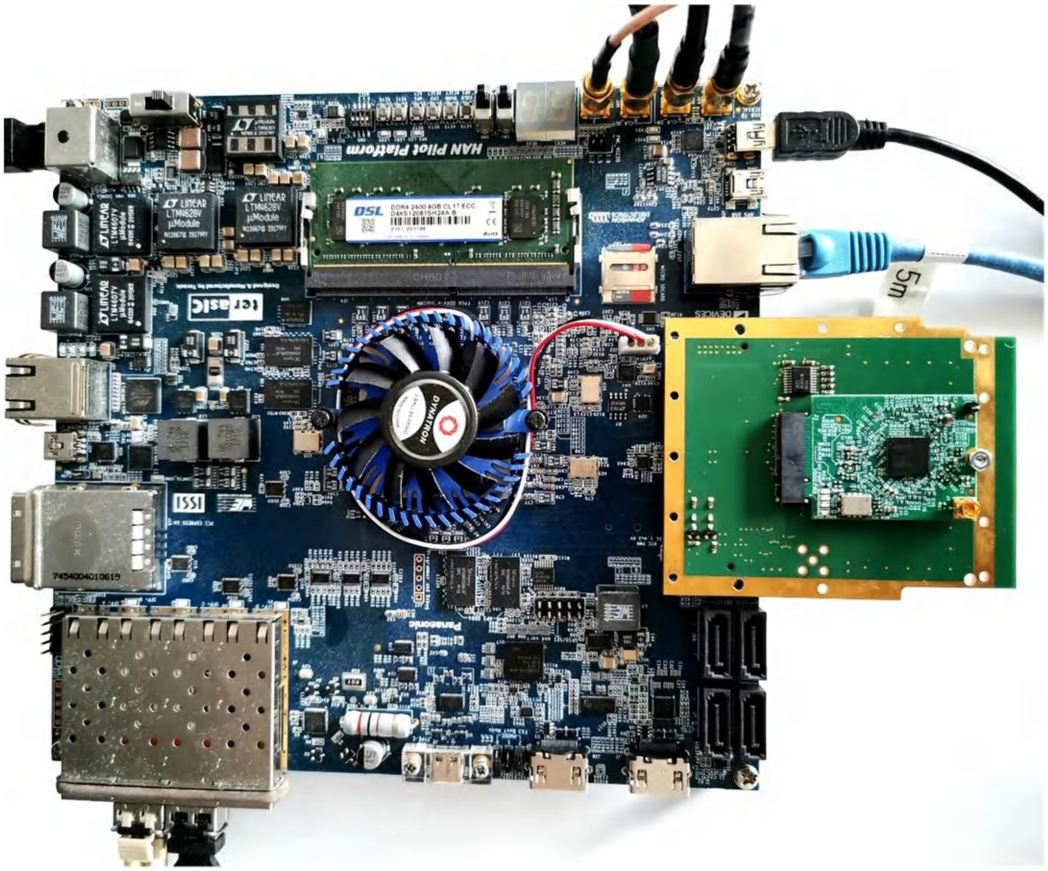


Figure 3-1. Lab Measurement Setup

## 3.1 FMC Adapter Board

A dedicated FMC adapter board is designed to directly attach the LMK5XXXXXS1 M2 module to the HAN Pilot platform in parallel. The adapter board contains an M2 socket, a number of passive components (such as pull-up/pull-down resistors and AC-coupling capacitors), and several level shifters. The top and bottom side of the adapter board are shown in Figure 3-2.

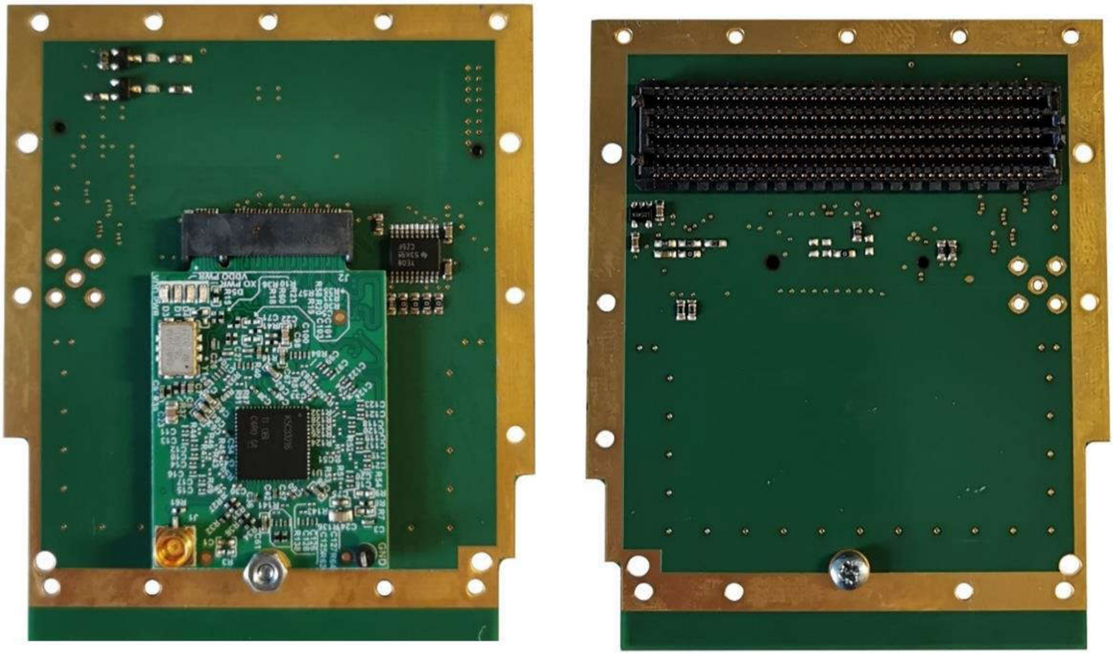
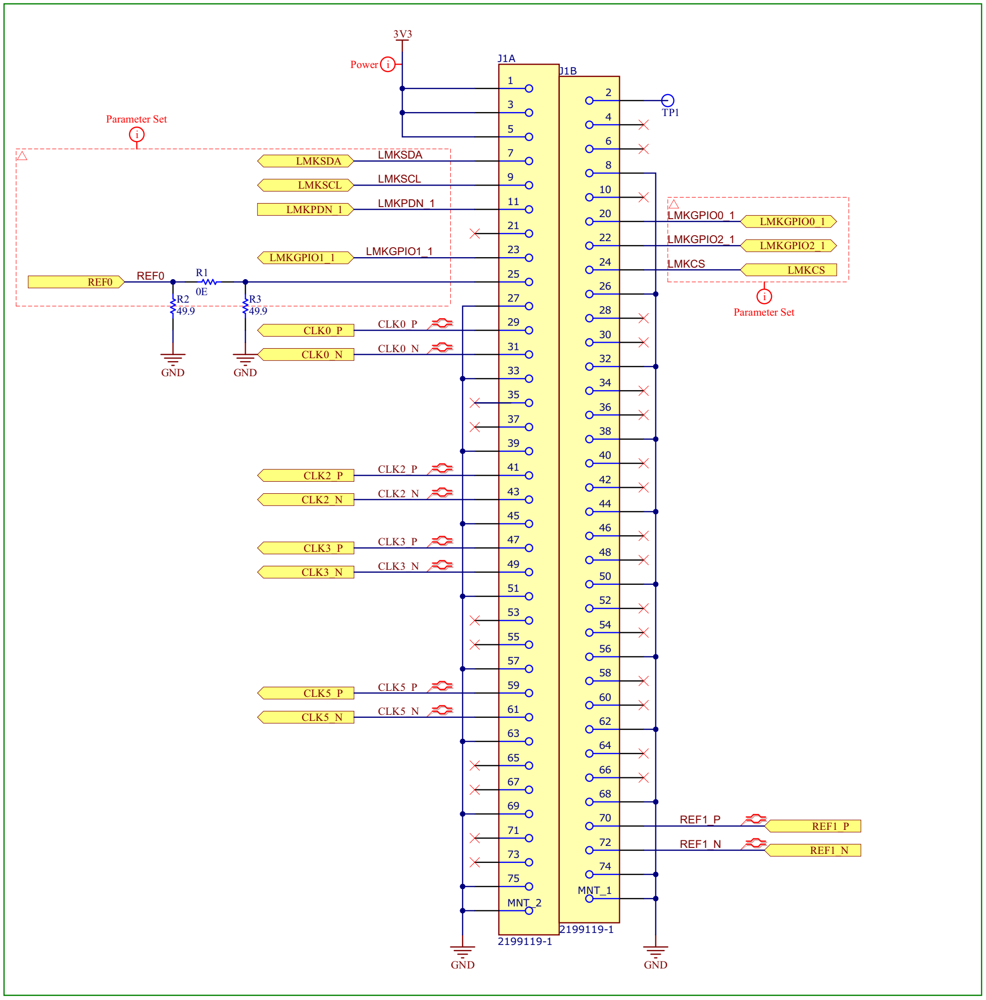


**Figure 3-2. FMC Adapter Board Containing the LMK5XXXXXS1 M2 Module - Top (Left) and Bottom (Right)**

In Figure 3-3, a connector schematic provides a more detailed view on the signals used from the LMK5XXXXXS1 M2 module, where REF0 (10MHz) is the primary input frequency for the network synchronizer. The secondary input, REF1 (156.25MHz), is the SyncE frequency extracted from the 10G Ethernet PHY port by the FPGA. OUT7 is the primary output frequency of the LMK5XXXXXS1, which is supplied directly to the PTP hardware clock. The value of the network timing PTP clock is adjusted by the PTP Stack as shown in Figure 1-3.

M2 connector



| FMC | | M2 | | PLL |
|---|---|---|---|---|
| GTBCLK0 | < | CLK2 | < | OUT0 |
| GTBCLK1 | < | CLK3 | < | OUT4 |
| CLK0_M2C | < | CLK0 | < | OUT2 |
| CLK1_M2C | < | CLK5 | < | OUT7 |
| CLK2_M2C | > | REF0 | > | REF0 |
| CLK3_M2C | > | REF1 | > | REF1 |

Variant: DNI

**Figure 3-3. FMC M2 Module Pins Used in FMC Adapter Board**

## 3.2 Compliance Test Setup

The compliance test setup is shown in Figure 3-4. In some test cases the Calnex Paragon-X is replaced with the Calnex Neo for improved nanosecond-level accuracy.

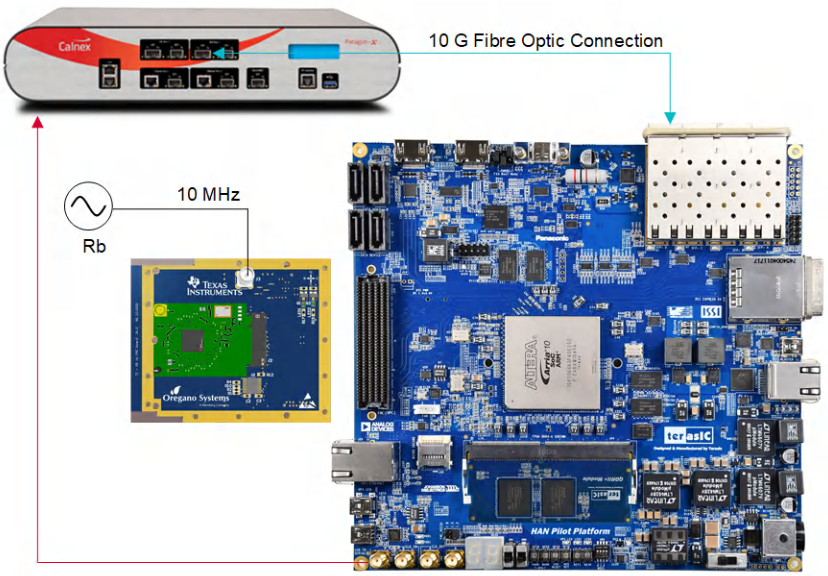


**Figure 3-4. Compliance Test Setup**

## 3.3 Compliance Test of Telecom Profile G.8275.1 - Full Timing Support

For networks with full timing support, the PTP stack on both Leader and Follower node devices are set to follow the default values of the PTP telecom profile G.8275.1:

- PTP domain number is 24.
- PTP event (Sync, Delay Request, Delay Respond) message rate is 16 packets per second.
- Announce message rate is 8 packets per second.
- Announce timeout is 3 missing messages.
- Ethernet address type is Multicast.
- Communication protocol is Layer 2.

The syn1588® PTP is configured as follows:

- The PI control loop parameters are set to *fast mode*, which is a setting optimized for minimal packet delay variation (PDV).
- The sample rate converter filter is activated
- The non-linear spike-prefilter is activated using the default windows size of 4 seconds
- The servo adjust rate is set to 1Hz

### *3.3.1 Transfer Characteristic*

The transfer characteristic of the servo noise transfer follows the specifications defined in the G.8283.2 for Class A and B. During the test, time information contained in the PTP event messages varied in a sinusoidal pattern with the frequency linearly increased by the Calnex Paragon X device used in testing. The result of this test is shown in a bode diagram in Figure 3-5 with the specification compliance limit lines illustrated in red.

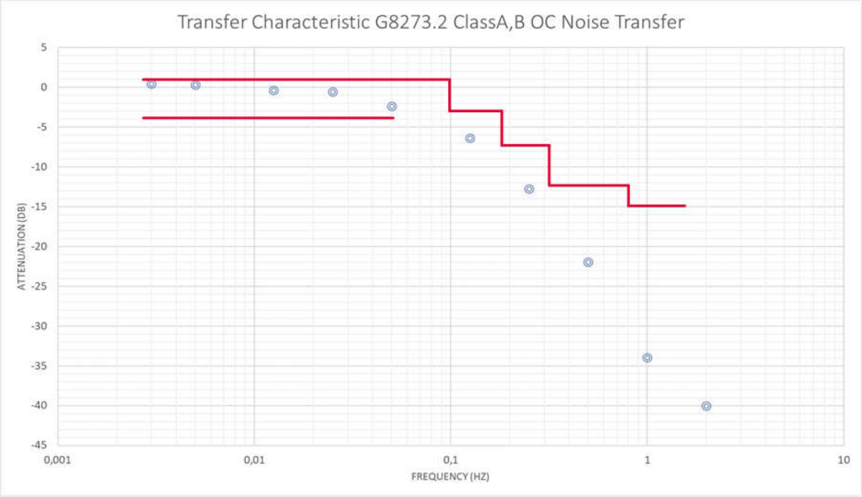


**Figure 3-5. Transfer Characteristic of PTP Servo for Networks With Full Timing Support**

### *3.3.2 Absolute Time Error*

In a second test, the accuracy (absolute time error) is measured by comparing the 1PPS signal of the PTP Grandmaster with the signal generated by the PTP implementation on the HAN Pilot Platform. As the Calnex Paragon X device can only measure 1PPS signals at a resolution of 2ns, a Calnex Paragon Neo is used for this measurement to increase the resolution.

The two 1PPS signals are offset-compensated to a level of less than 2ns, accounting for different coaxial cable lengths and varying transmission delays within the respective PCS PMA units.

The results in show an the comparison from two way time error with a signal spread of less than ± 2ns exceeding the limits for the Class D performance in a network with full timing support. The results are shown in Figure 3-6. The time error when measured on the 1PPS signal shows the results are similar without any high pass filtering as shown in Figure 3-6 and Figure 3-6.

The minimum time interval error or MTIE is also very well behaved within limits, as shown in Figure 3-9

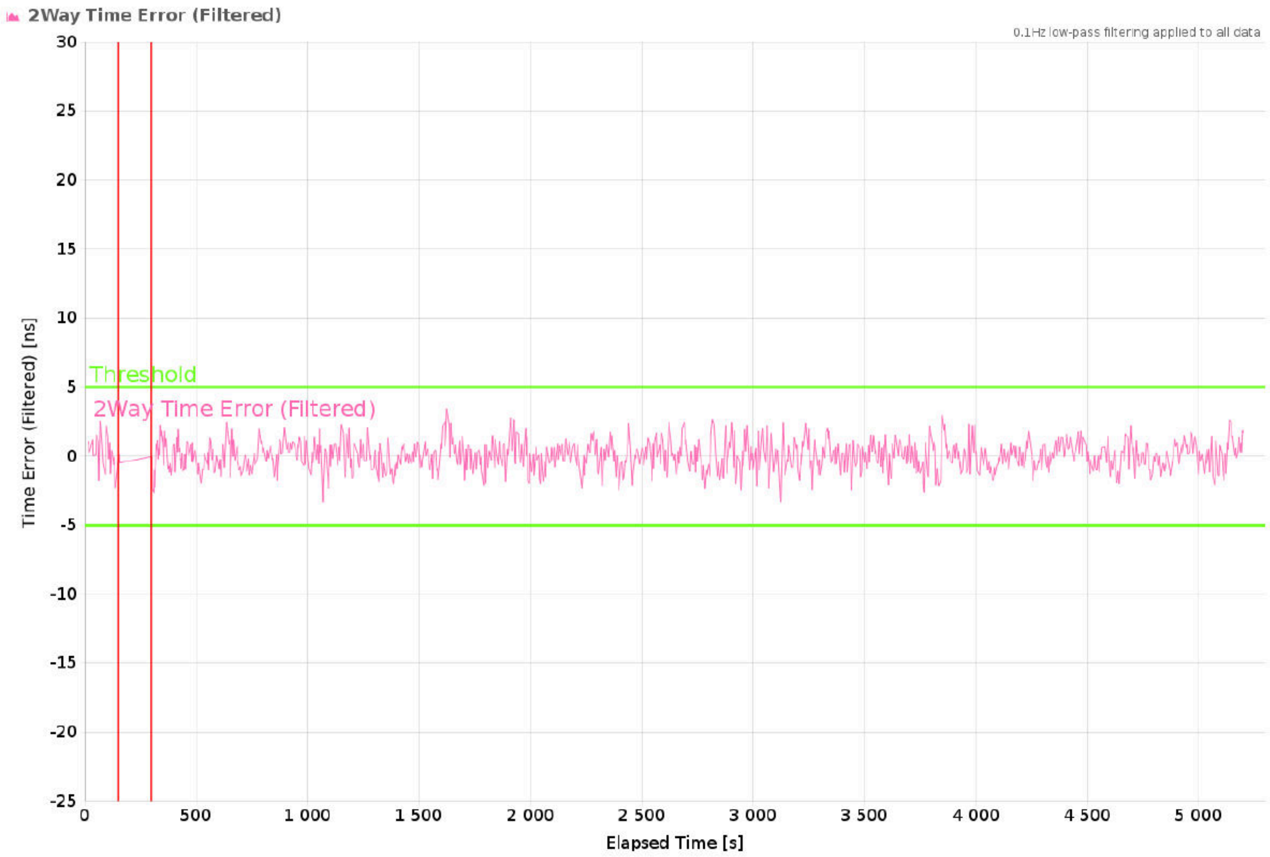
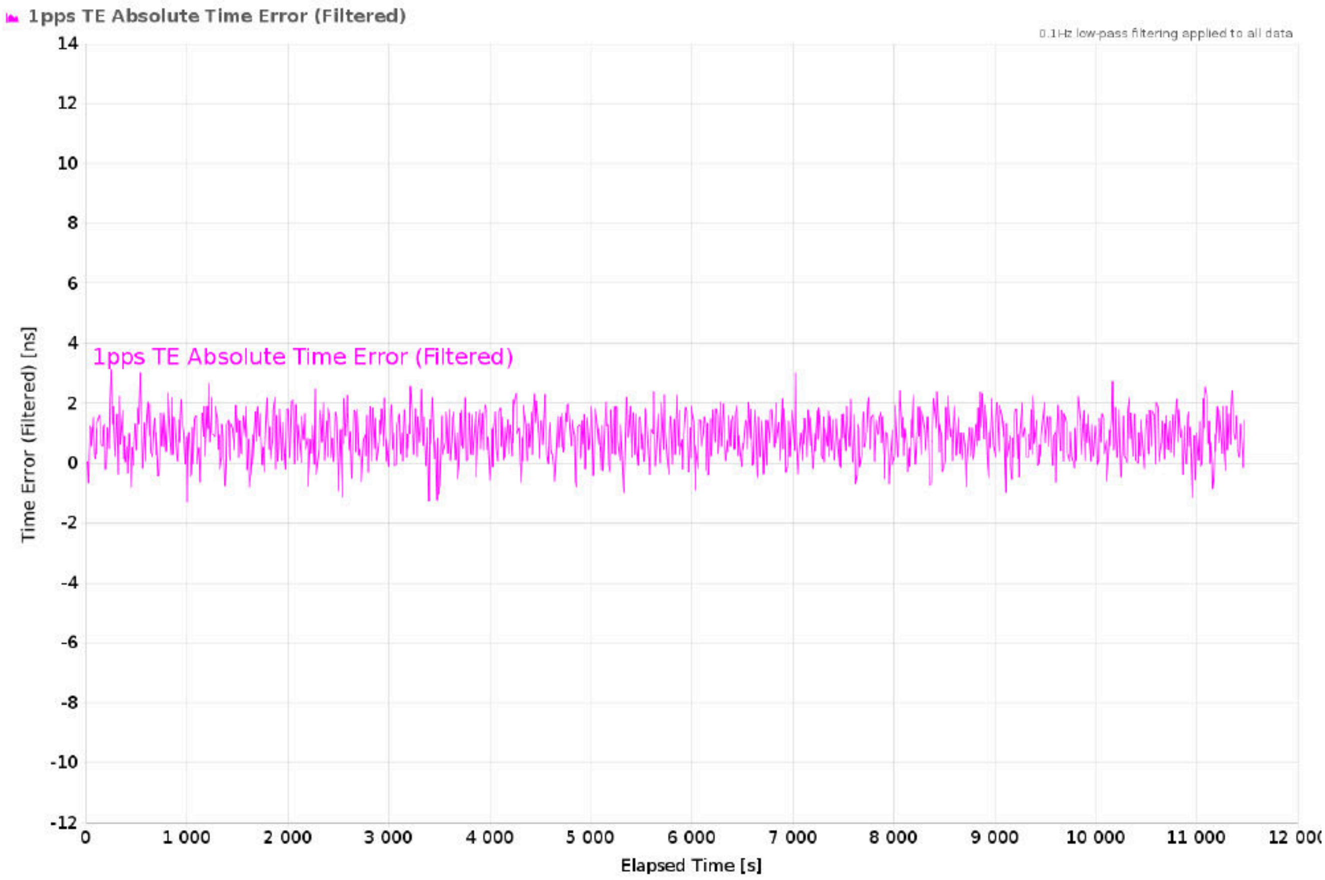
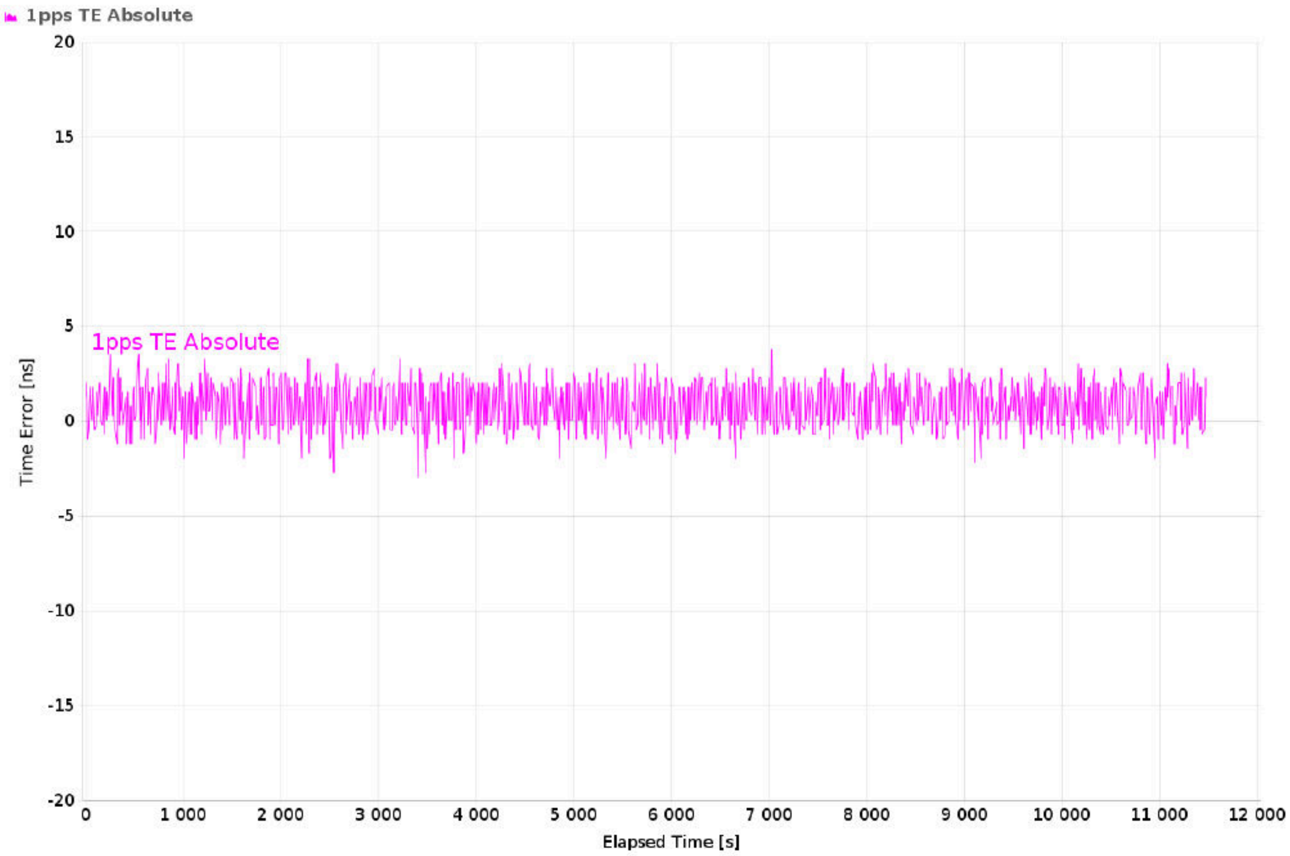


**Figure 3-6. Two Way Time Error Filtered with Class D Limit Lines (±5ns), Typical Measurement**



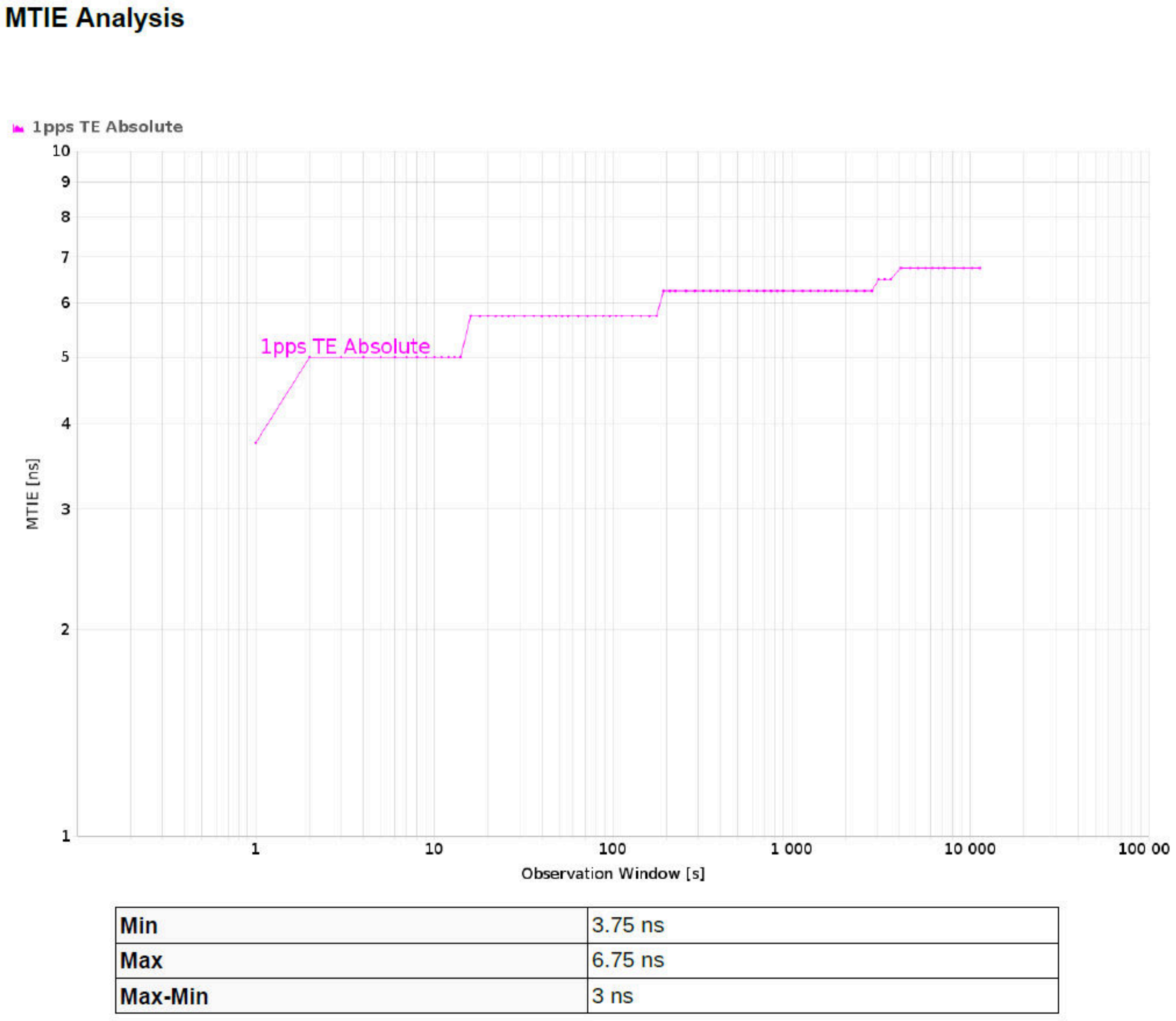
| | |
|---|---|
| **Mean** | 0.914 ns |
| **Min** | -1.306 ns |
| **Max** | 3.12 ns |
| **Max-Min** | 4.426 ns |

**Figure 3-7. 1PPS Absolute Time Error Filtered with 0.1Hz High-Pass Filter, Typical Measurement**



| | |
|---|---|
| **Mean** | 0.914 ns |
| **Min** | -2.952 ns |
| **Max** | 3.798 ns |
| **Max-Min** | 6.75 ns |

**Figure 3-8. 1PPS Absolute Time Error (No Filter), Typical Measurement**



MTIE Analysis

| | |
|---|---|
| **Min** | 3.75 ns |
| **Max** | 6.75 ns |
| **Max-Min** | 3 ns |

**Figure 3-9. MTIE Analysis, Typical Measurement**

### *3.3.3 Lock Time*

In addition to compliance, the locking behavior of the syn1588® PTP Stack is investigated in the compliance test. The overall lock time can be divided into two intervals. The first interval starts by invoking the PTP Stack, covering the initial message processing that the PTP stack requires to complete prior to accepting a PTP Grandmaster as a time source. The PTP Stack processes the data contained in a series of consecutive PTP Announce messages. The total number of consecutive messages the PTP stack has to consider before accepting a PTP Grandmaster, as well as the rate of the PTP Stack, define the time span. The parameters are user-configurable, with most PTP profiles defining both subranges and default values for each profile.

The second interval depends on the initial synchronization algorithm of the PTP Stack, in combination with the capabilities of the underlying hardware. The design implemented in the syn1588® PTP Stack is described above in the Synchronization Algorithm section. The second interval is most effectively measured by evaluating the data in the log file of the PTP Stack. The respective section of a typical startup process is given in the log output below.

The first line in the log output indicates the point in time where the PTP Stack starts using time information from a PTP Grandmaster. The large offset (-1646305347758870528ns) causes the PTP Stack to asynchronously set the PTP hardware clock indicated by the log output *do epoch jump to …* on line 2.

The residual offset (in -10ms) is shown in subsequent lines. The PTP Stack calculates the frequency offset using a sequence of downstream PTP event messages. After collecting enough data, the PTP Stack adjusts both the frequency and offset synchronously. As a final step, the PTP Stack enables the PI control loop, which can be seen in the last 5 lines shown in the log output below.

In this test, the length of the second interval is calculated using the system time stamps of the first and last line of the log output, respectively.

***** this is the following text inside an equation block? If this is an equation, please create the equation in the Word equation editor and then upload using the TI Equation tool *****

$$12{:}01{:}57.09895 - 12{:}01{:}53.57872 \cong 3.5\text{s} \qquad (1)$$

If the default values for the Announce rate and timeout are used, the duration of the first interval is less than 1 second.

```
12:01:53.57872 TimestampStats<meanTime=1996-02-01 06:31:16 ptp, pathDelay=577ns, offset=
   -1646305347758870528ns>
12:01:53.57875 do epoch jump to 2022-03-03 12:02:30 ptp
12:01:54.32878 do no adjust as filters are not ready
12:01:54.81291 TimestampStats<meanTime=2022-03-03 12:02:31 ptp, pathDelay=596ns, offset=-10985308ns>
12:01:54.81294 change to CALIBRATE state
12:01:54.86872 TimestampStats<meanTime=2022-03-03 12:02:31 ptp, pathDelay=591ns, offset=-10985333ns>
12:01:54.91748 TimestampStats<meanTime=2022-03-03 12:02:31 ptp, pathDelay=590ns, offset=-10985367ns>
12:01:54.95272 TimestampStats<meanTime=2022-03-03 12:02:31 ptp, pathDelay=570ns, offset=-10985410ns>
12:01:55.03707 TimestampStats<meanTime=2022-03-03 12:02:31 ptp, pathDelay=586ns, offset=-10985460ns>
12:01:55.09210 TimestampStats<meanTime=2022-03-03 12:02:31 ptp, pathDelay=580ns, offset=-10985494ns>
12:01:55.17129 TimestampStats<meanTime=2022-03-03 12:02:31 ptp, pathDelay=588ns, offset=-10985541ns>
12:01:55.26151 TimestampStats<meanTime=2022-03-03 12:02:31 ptp, pathDelay=575ns, offset=-10985619ns>
12:01:55.32995 do no adjust as filters are not ready
12:01:55.34612 TimestampStats<meanTime=2022-03-03 12:02:31 ptp, pathDelay=591ns, offset=-10985666ns>
12:01:55.40650 TimestampStats<meanTime=2022-03-03 12:02:31 ptp, pathDelay=577ns, offset=-10985701ns>
12:01:55.46685 TimestampStats <meanTime=2022-03-03 12:02:31 ptp, pathDelay=578ns,
offset=-10985745ns>
12:01:55.50951 TimestampStats <meanTime=2022-03-03 12:02:31 ptp, pathDelay=576ns,
offset=-10985782ns>
12:01:55.60322 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=592ns,
offset=-10985832ns>
12:01:55.64053 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=576ns,
offset=-10985870ns>
12:01:55.77892 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=578ns,
offset=-10985957ns>
12:01:55.83618 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=578ns,
offset=-10985998ns>
12:01:55.89710 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=580ns,
offset=-10986037ns>
12:01:55.94715 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=572ns,
offset=-10986066ns>
12:01:56.02576 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=591ns,
offset=-10986119ns>
12:01:56.11254 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=582ns,
offset=-10986174ns>
12:01:56.16287 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=581ns,
offset=-10986208ns>
12:01:56.22994 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=587ns,
offset=-10986256ns>
12:01:56.30856 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=593ns,
offset=-10986300ns>
12:01:56.32965 do no adjust as filters are not ready
12:01:56.36520 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=591ns,
offset=-10986335ns>
12:01:56.42143 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=585ns,
offset=-10986379ns>
12:01:56.46719 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=582ns,
offset=-10986415ns>
12:01:56.56008 TimestampStats <meanTime=2022-03-03 12:02:32 ptp, pathDelay=588ns,
offset=-10986468ns>
12:01:56.65093 TimestampStats <meanTime=2022-03-03 12:02:33 ptp, pathDelay=579ns,
offset=-10986542ns>
12:01:56.70391 TimestampStats <meanTime=2022-03-03 12:02:33 ptp, pathDelay=573ns,
offset=-10986581ns>
12:01:56.77593 TimestampStats <meanTime=2022-03-03 12:02:33 ptp, pathDelay=585ns,
offset=-10986615ns>
```

```
12:01:56.84436 TimestampStats <meanTime=2022-03-03 12:02:33 ptp, pathDelay=583ns,
offset=-10986664ns>
12:01:56.93805 TimestampStats <meanTime=2022-03-03 12:02:33 ptp, pathDelay=588ns,
offset=-10986722ns>
12:01:57.01258 TimestampStats <meanTime=2022-03-03 12:02:33 ptp, pathDelay=570ns,
offset=-10986790ns>
12:01:57.01261 adjust drift -6.65936e-07s/s
12:01:57.01264 Adjust rate by -665.936 ns/s (speeding up)
12:01:57.01269 compensate Offset -0.0109868s
12:01:57.01274 change to COLLECT_DATA state
12:01:57.09895 TimestampStats <meanTime=2022-03-03 12:02:33 ptp, pathDelay=577ns, offset=-10ns>
```

## 3.4 Compliance Test of Telecom Profile G.8275.2 - Partial Timing Support

For the test case purpose of networks with partial timing support, the PTP stacks on both leader and follower node devices are configured using default values of the PTP telecom profile G.8275.2.

The PTP event message rate details are listed:

- PTP domain number is 44.
- PTP event message rate is 64 packets per second.
- Announce message rate is 1 packet per second
- Announce timeout is 3 missing messages.
- Ethernet address type is unicast.
- Communication protocol is IPv4

The syn1588® PTP was configured as follows:

- The control loop parameters are set to slow mode, a setting optimized for high PDVs
- Sample rate converter filter is activated
- The non-linear spike prefilter is deactivated
- The non-linear lucky packet file is enabled with a packet buffer of 4069 packets and a median filter width of 3 packets.
- The clock is adjusted once every 8 seconds.

Initial test runs reveal that the syn1588® PTP Stack required further optimization to cope with the PDVs introduced by the Calnex system. A simple test is performed to analyze the noise level upstream and downstream of the signal. The ToD clock of the device under test is synchronized externally using the 1PPS signal provided by the Calnex. As soon as the device is sufficiently synchronized and test version of PTP Stack is invoked, the test processes all PTP messages and calculates the offset and path delay accordingly, while leaving the hardware clock unadjusted. Using the time stamp data in the log file, the impairment generated by the Calnex Paragon X can be analyzed.

In Figure 3-10 the packet delay variations of the Sync messages (T2 – T1) are shown, while in Figure 3-11 the same data for the Delay Request Messages is provided (T4 – T3). The raw offset is calculated using the following formula:

$$Clock\_Offset = \frac{(T_2 - T_1) - (T_4 - T_3)}{2} \tag{2}$$

The resulting graph is shown in Figure 3-12. A detailed view of the initial phase of the impairment is provided in Figure 3-13.

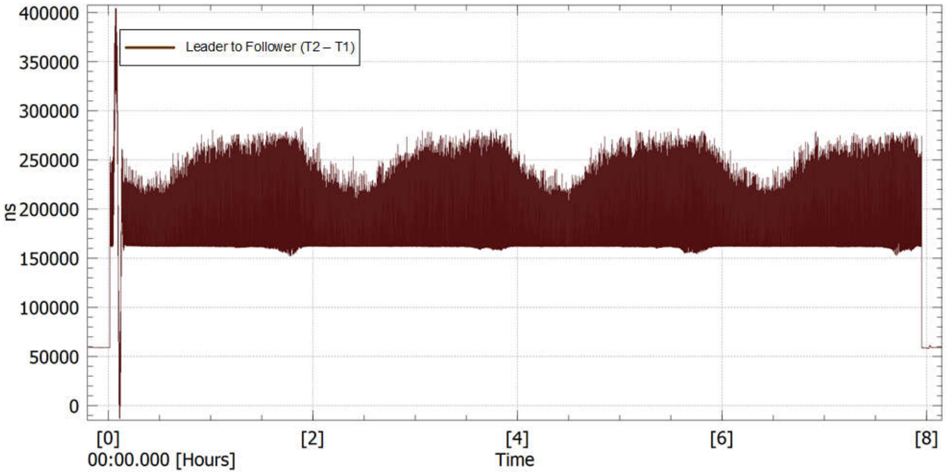


**Figure 3-10. Leader to Follower Delay (T2-T1) During Impairment**

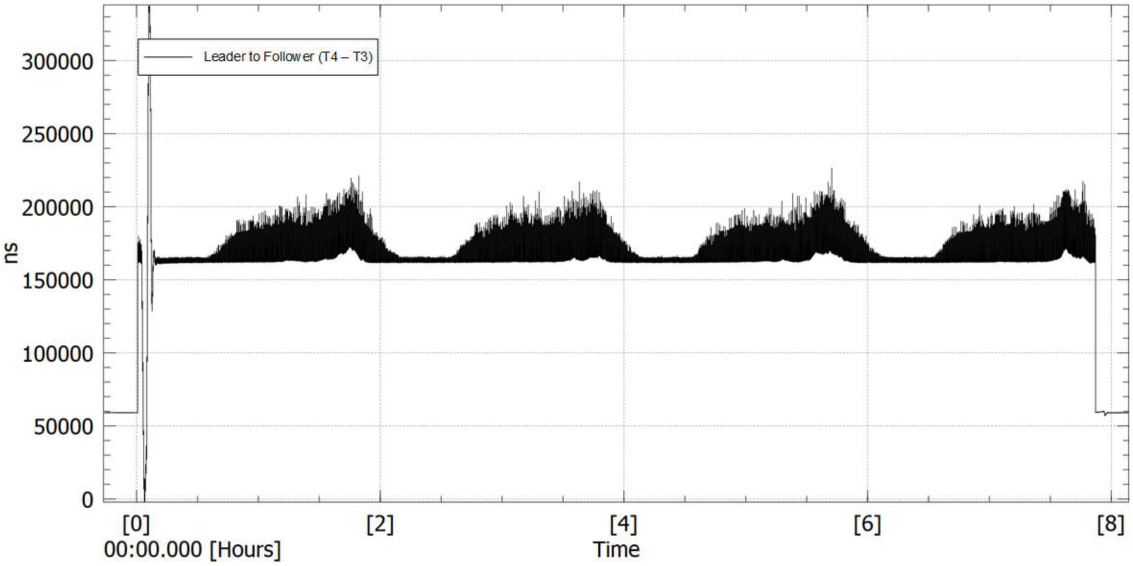


**Figure 3-11. Follower to Leader Delay (T4-T3) During Impairment**

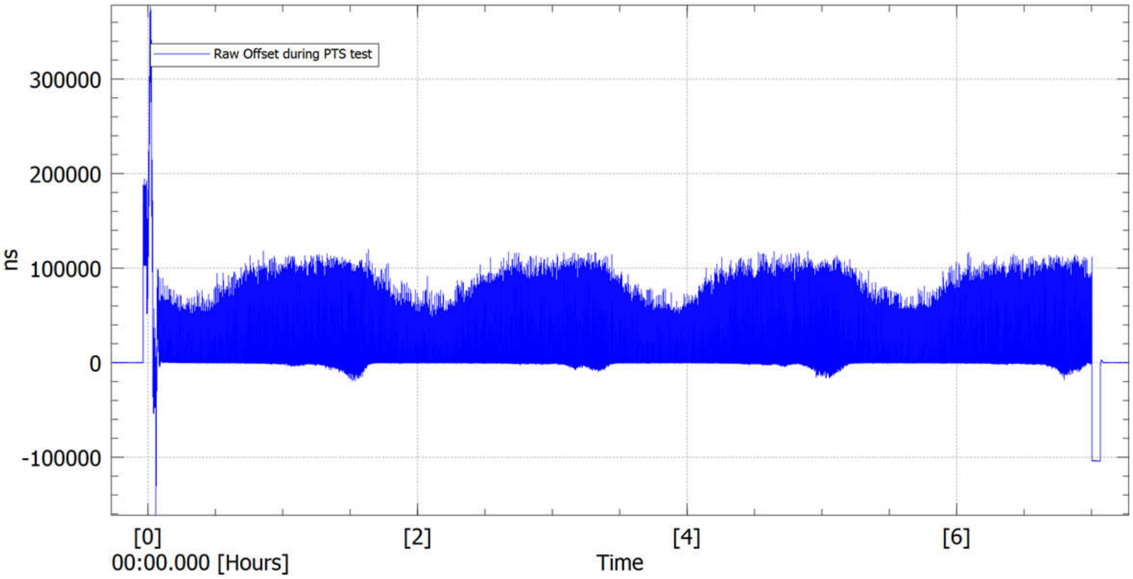


**Figure 3-12. Offset Calculated Using (T1 … T4) During Impairment**

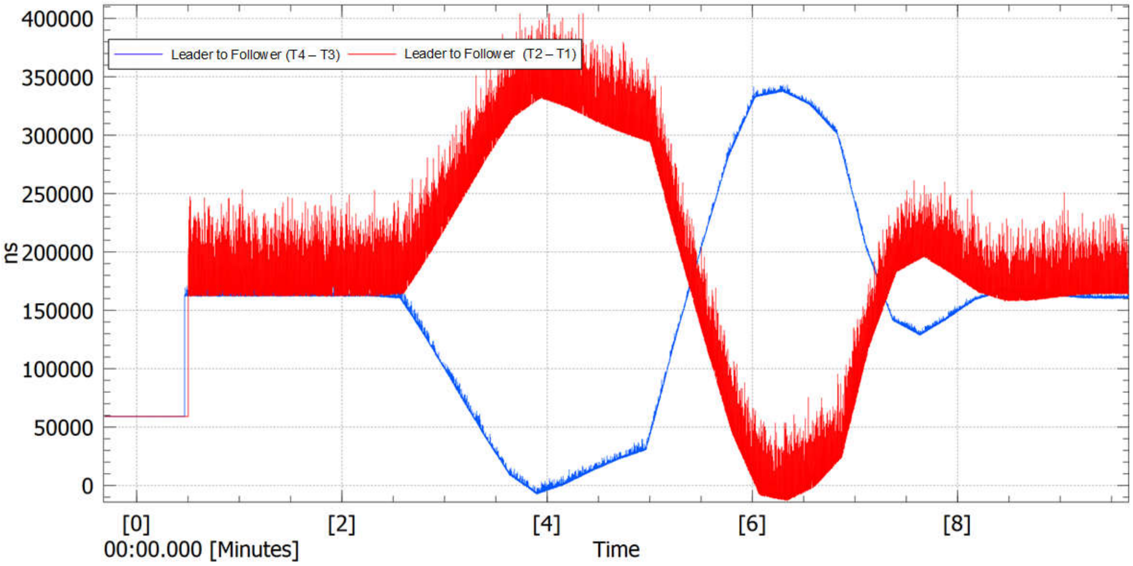
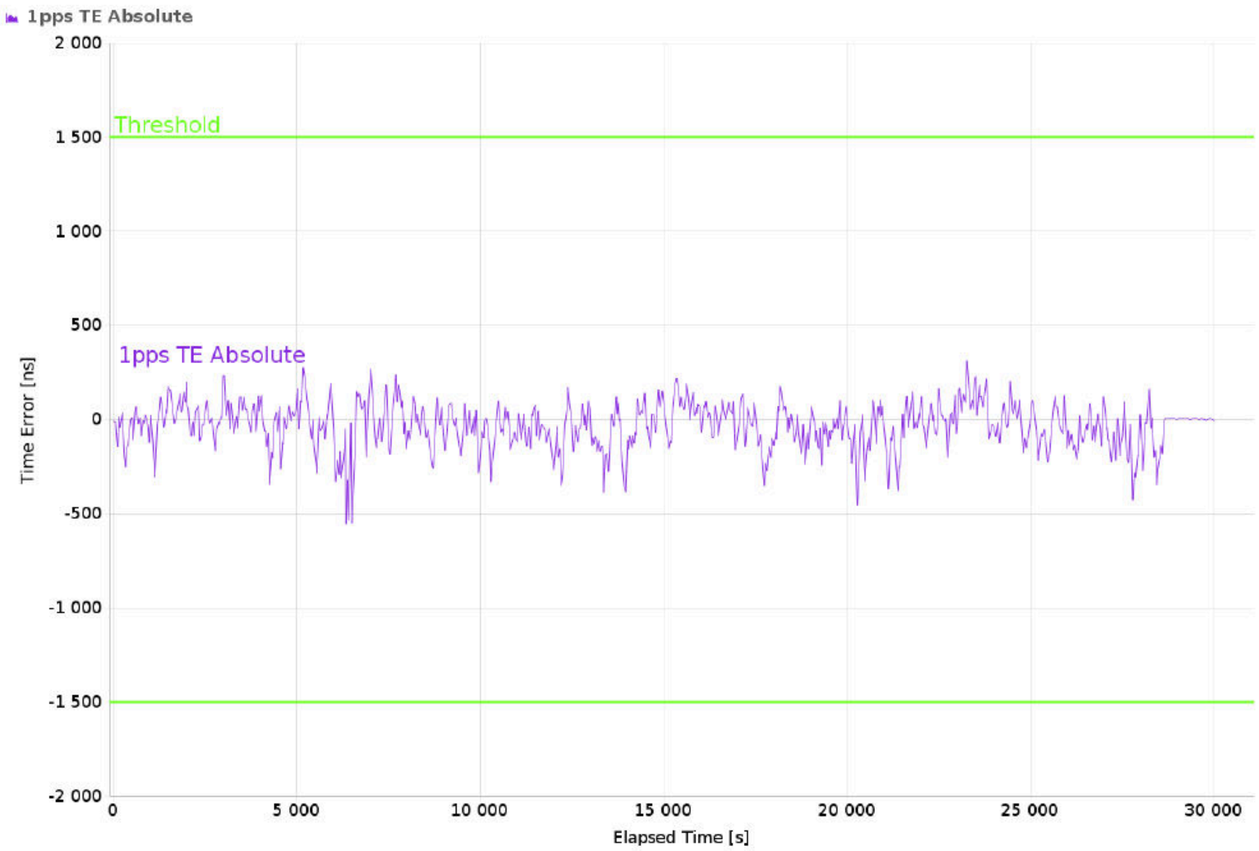


**Figure 3-13. Upstream and Downstream Delay Variations During Impairment (Capture is Zoomed in to Show the Beginning of the Test Cycle)**

To cope with changing conditions in the environment or setup, the PTP stack is configured to adjust the hardware clock once every 8 seconds. Additionally, the search window of the lucky packet filter is extended to 4069 packets. The test setup is as follows:

- Prior to enabling the test, the PTP stack is allowed to settle for 10 to 15 minutes. This long settling time is attributed to the servo parameter setting.
- The impairment is enabled and the 1PPS signal of the Leader and the Follower are monitored.

The results are shown in Figure 3-14. The test is repeated multiple times, yielding similar results.



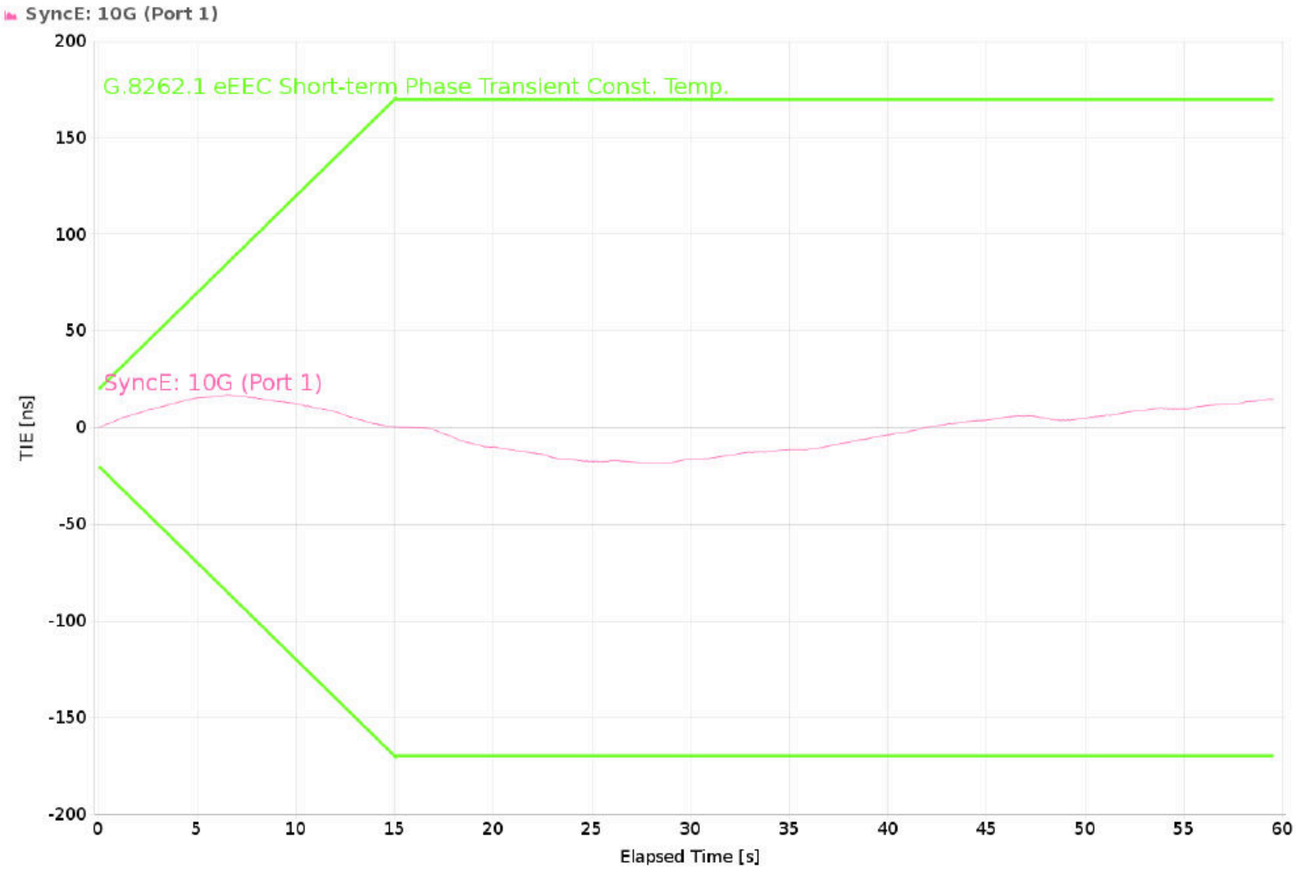
| | |
|---|---|
| **Mean** | -34.494 ns |
| **Min** | -554.702 ns |
| **Max** | 314.048 ns |
| **Max-Min** | 868.75 ns |

**Figure 3-14. Absolute Time Error (Comparison of 1PPS Signals) for Partial Timing Support Test, Typical Measurement**

## 3.5 Compliance Test of Telecom Profile G.8262.1 - SyncE Transient

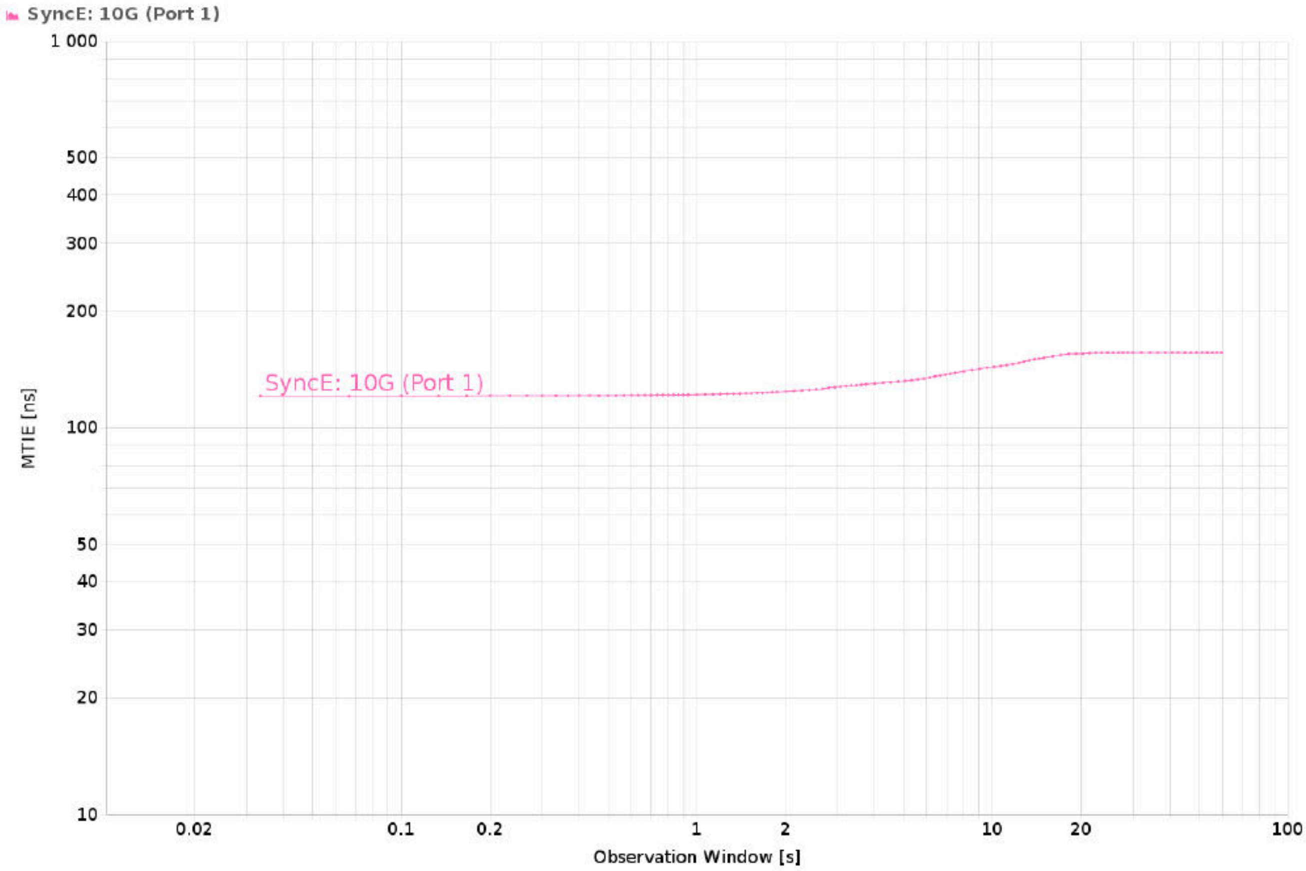
The short-term transient response refers to the time accuracy error occurring when a clock switches over from one input reference to another. A reference switch over must adhere to the limits and conditions for SyncE short term transients as described in G.8262.1

The resulting graphs are shown in Figure 3-15 and Figure 3-16.



| | |
|---|---|
| **Mean** | 18.119 ns |
| **Min** | 0 ns |
| **Max** | 33.074 ns |
| **Max-Min** | 33.074 ns |
| **Rate** | 30.00/second |

**Figure 3-15. SyncE Transient TIE**



| | |
|---|---|
| **Min** | 120.76 ns |
| **Max** | 156.348 ns |
| **Max-Min** | 35.588 ns |

**Figure 3-16. SyncE Transient MTIE**

# 4 PTP System Application

The PTP clock can be adjusted for 5G wireless applications by applying the same DCO algorithm on the LMK5CxxxxAS1 for DPLL3 as shown in Figure 4-1. The configuration with LMK5CxxxxAS1 is used to achieve the telecommunication accuracy of Class C performance and above while providing the best radio clock jitter performance in the industry, with a maximum jitter of 57fs (12kHz to 20MHz) for 491.52MHz output clocks. With the flexibility provided by the LMK5XXXXXS1 for internal feedback between DPLL and APLL domains, the domains can be cascaded depending on whether a separate PTP domain or separate SyncE domain is required or whether the hybrid PTP and SyncE are combined.

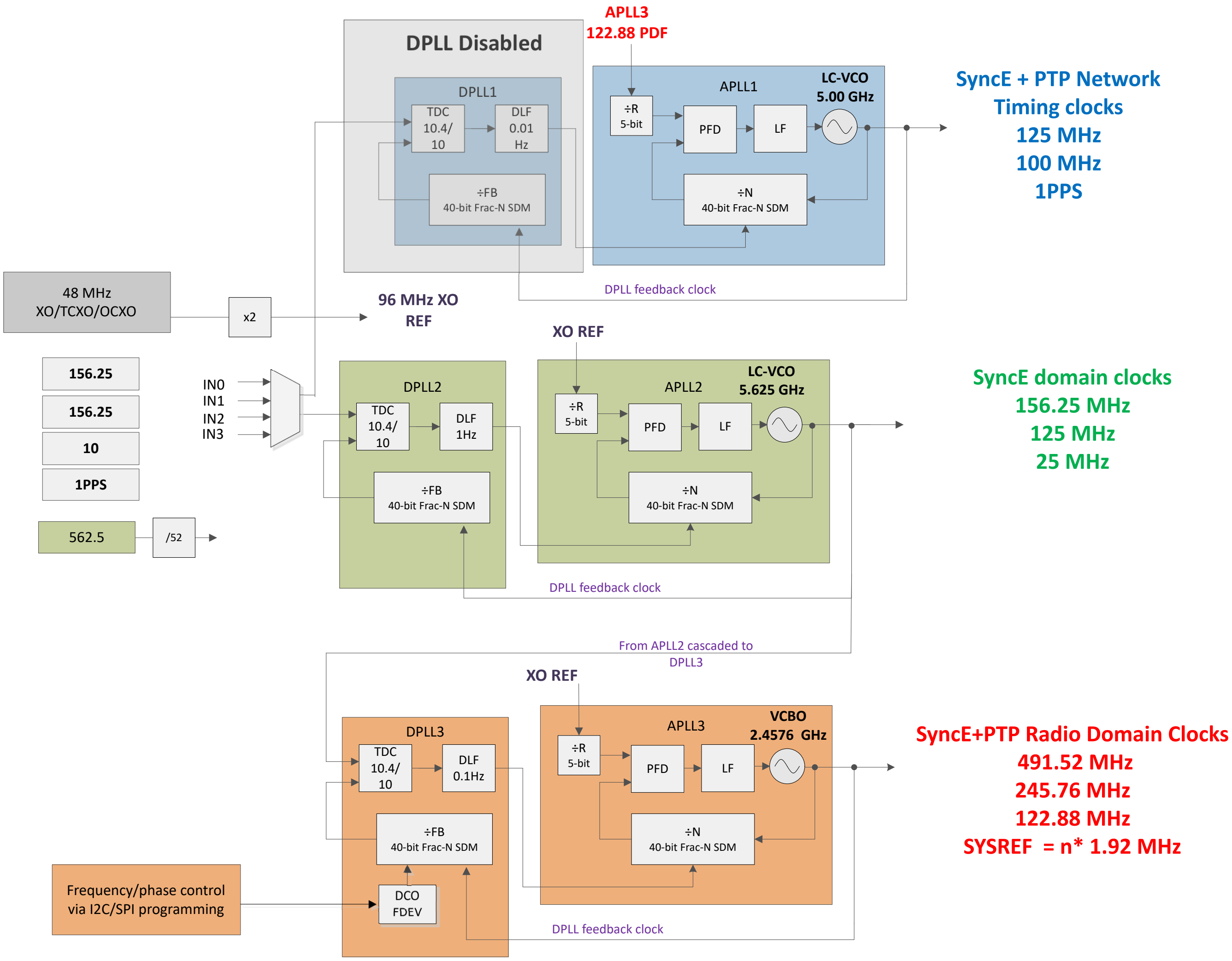


**Figure 4-1. PTP and SyncE for 5G Applications**

## 5 Additional Development

The initial settling of the syn1588® PTP Stack can be optimized more by taking the quality of the local oscillator into account, which improves the settling behavior under network impairments. Furthermore, the holdover performance can also be further optimized by taking the features of the network synchronizer into account. Finally, extended tests with SyncE capable devices are planned to investigate both the holdover and switching performance of the complete PTP system using the LMK5XXXXXS1.

# 6 Conclusion

The combination of the syn1588® on the Arria® 10 with the LMK5XXXXXS1 is compliant with the telecommunication accuracies of Class A (100ns), Class B (70ns), Class C (30ns), and even Class D (5ns). The 1PPS signal does not deviate more than 2ns. A high-grade local oscillator offering a short-term stability of less than 1ns over 5s is mandatory to reach Class D. Additionally, all measurements are made using a direct connection between two PTP device rather than connecting one or more Boundary Clocks or Transparent Clocks.

For networks with partial timing support, the configuration of the PTP stack is optimized to handle PDVs in excess of 230μs while staying within the requested boundaries of ±1.5μs with a headroom of 500ns.

## 7 References

1. Texas Instruments, *LMK5C33216A Product Folder*
2. Texas Instruments, *LMK5C33216AS1 Product Folder*
3. International Telecommunication Union (ITU), *G.8275.1 : Precision time protocol telecom profile for phase/time synchronization with full timing support from the network*
4. International Telecommunication Union (ITU), *G.8275.2 : Precision time protocol telecom profile for time/phase synchronization with partial timing support from the network*
5. terasIC, *Arria® 10 Han Pilot Platform*
6. Texas Instruments, *TICS Pro Software*
7. International Telecommunication Union (ITU), G.8273.2 : Timing characteristics of telecom boundary clocks and telecom time slave clocks for use with full timing support from the network

# IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to TI's Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265