OCP/TAP Dec 2021

Platform Clock Offset Measurement Using PTM

Christopher Hall (christopher.s.hall@intel.com)

christopher.s.hall@intel.com

NOTICES & DISCLAIMERS

- This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.
- No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.
- Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.
- The products and services described may contain defects or errors known as errata which may cause deviations from published specifications. Current characterized errata are available on request. No product or component can be absolutely secure.
- Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm.
- Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No product or component can be absolutely secure. Check with your system manufacturer or retailer or learn more at intel.com.
- Intel, the Intel logo, Pentium, Celeron, Atom, Core, Xeon, Movidius and others are trademarks of Intel Corporation in the U.S. and/or other countries.
  *Other names and brands may be claimed as the property of others.
- All logos, photos, etc. used in this presentation are the property of their respective copyright owners.

-

Agenda

- Problem Statement
- Discuss PTM Protocol
  - Overview of Measurement Platform
- Test Methodology
- Measurement Results
- Next Steps

christopher.s.hall@intel.com

Problem Statement

- Background: PTP protocol software (e.g. Linux PTP) enables synchronization of the network device clock with the GM
- The device clock is not directly available to software
  - Issuing a read to the device requires a read across the PCIe bus in the case of a discrete card
  - The latency is 100s of nanoseconds
- The Linux PTP solution uses PHC2SYS to synchronize the system clock – that is typically based on a CPU timer – to the device clock
- PHC2SYS by default uses a software method to compute the offset between the clocks which introduces inaccuracy

christopher.s.hall@intel.com

PHC2SYS Offset Computation in Software

S_{start} = Read System Time

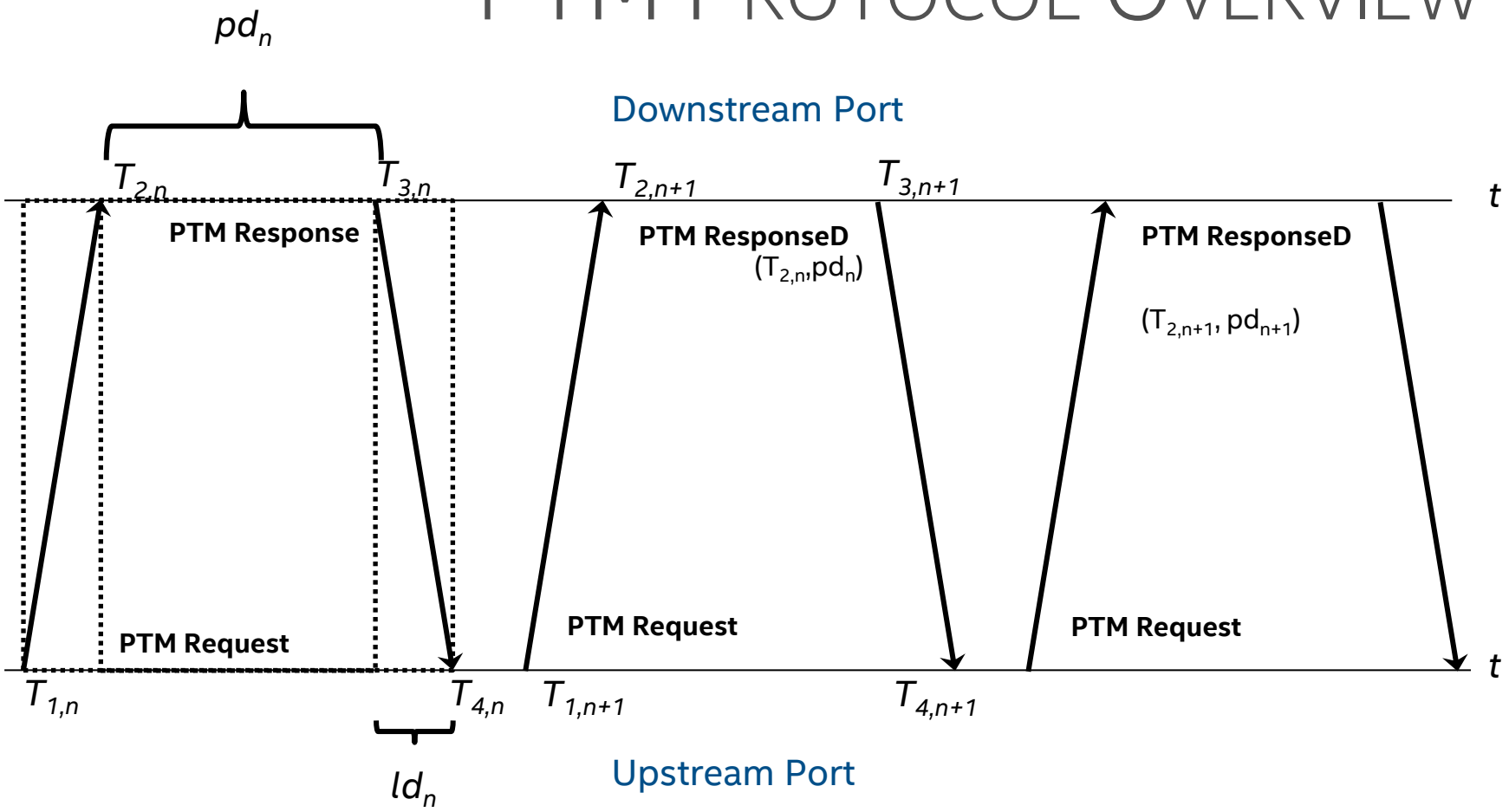
Device = Read Device Time

S_{end} = Read System Time

$$Offset = Device - (S_{start} + S_{end})/2$$

PTM Protocol and Test Methodology

PTM Protocol Overview

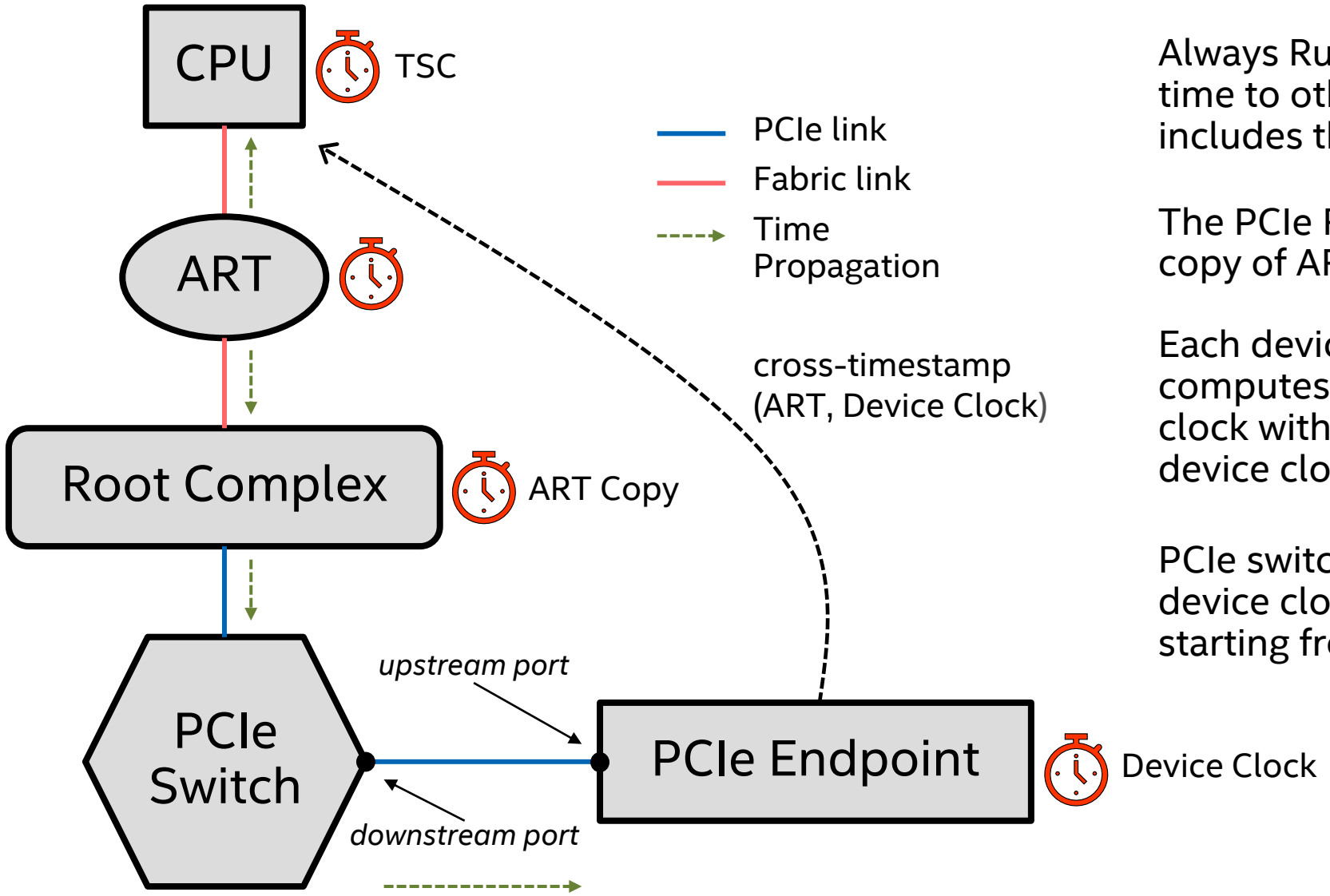


Propagation Delay (pd_n) = $T_{3,n}$-$T_{2,n}$

Compute "Link" Delay (ld_n) = [($T_{4,n}$-$T_{1,n}$) - pd_n]/2

Compute Upstream / Downstream Offset = ($T_{1,n}$ + ld_n) – $T_{2,n}$

Intel PCIe Clock Topology



Always Running Timer (ART) provides time to other devices on the system – includes the CPU and PCIe peripherals

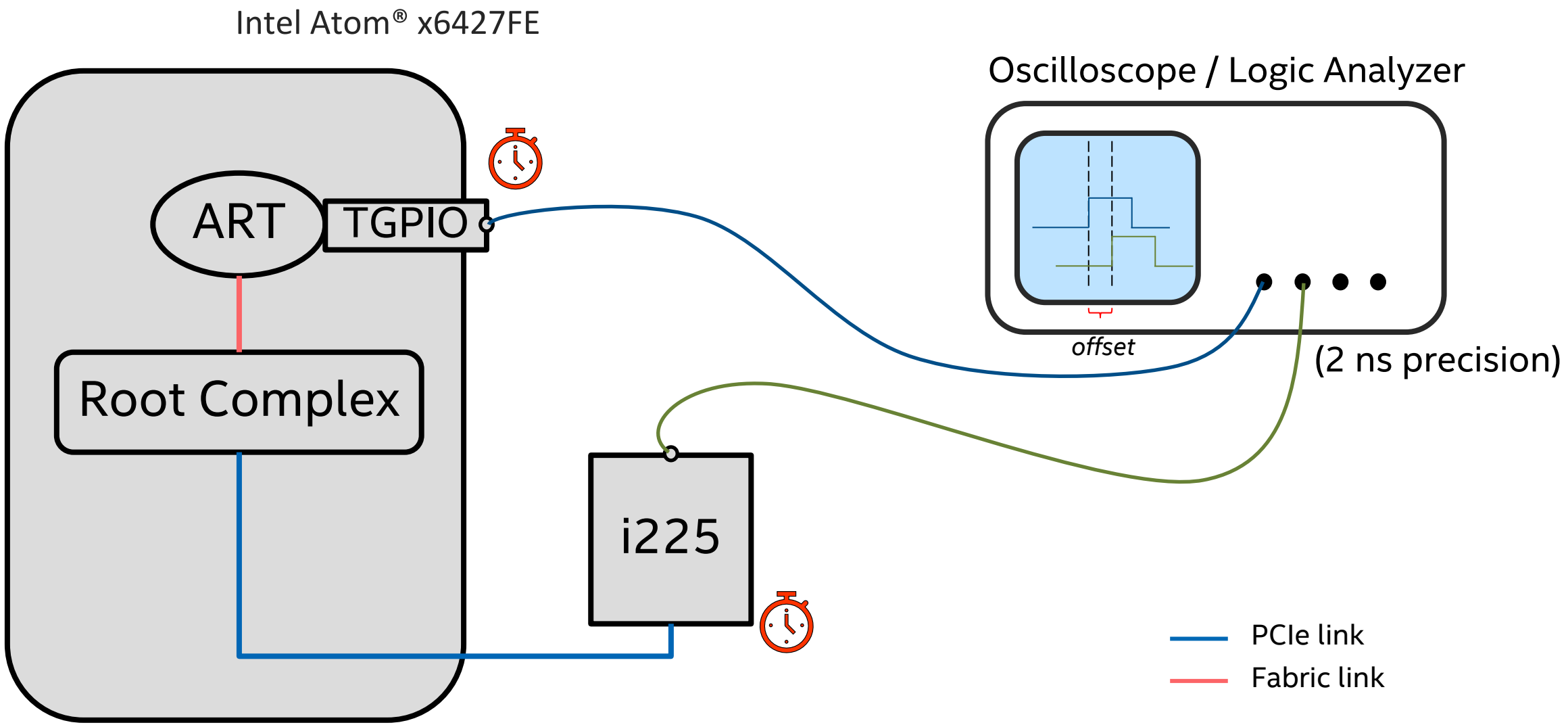
The PCIe Root Complex implements a copy of ART

Each device (e.g. switch, endpoint) computes the offset of its local device clock with respect to the upstream device clock using PTM

PCIe switches propagate the upstream device clock to downstream devices starting from the root complex

christopher.s.hall@intel.com

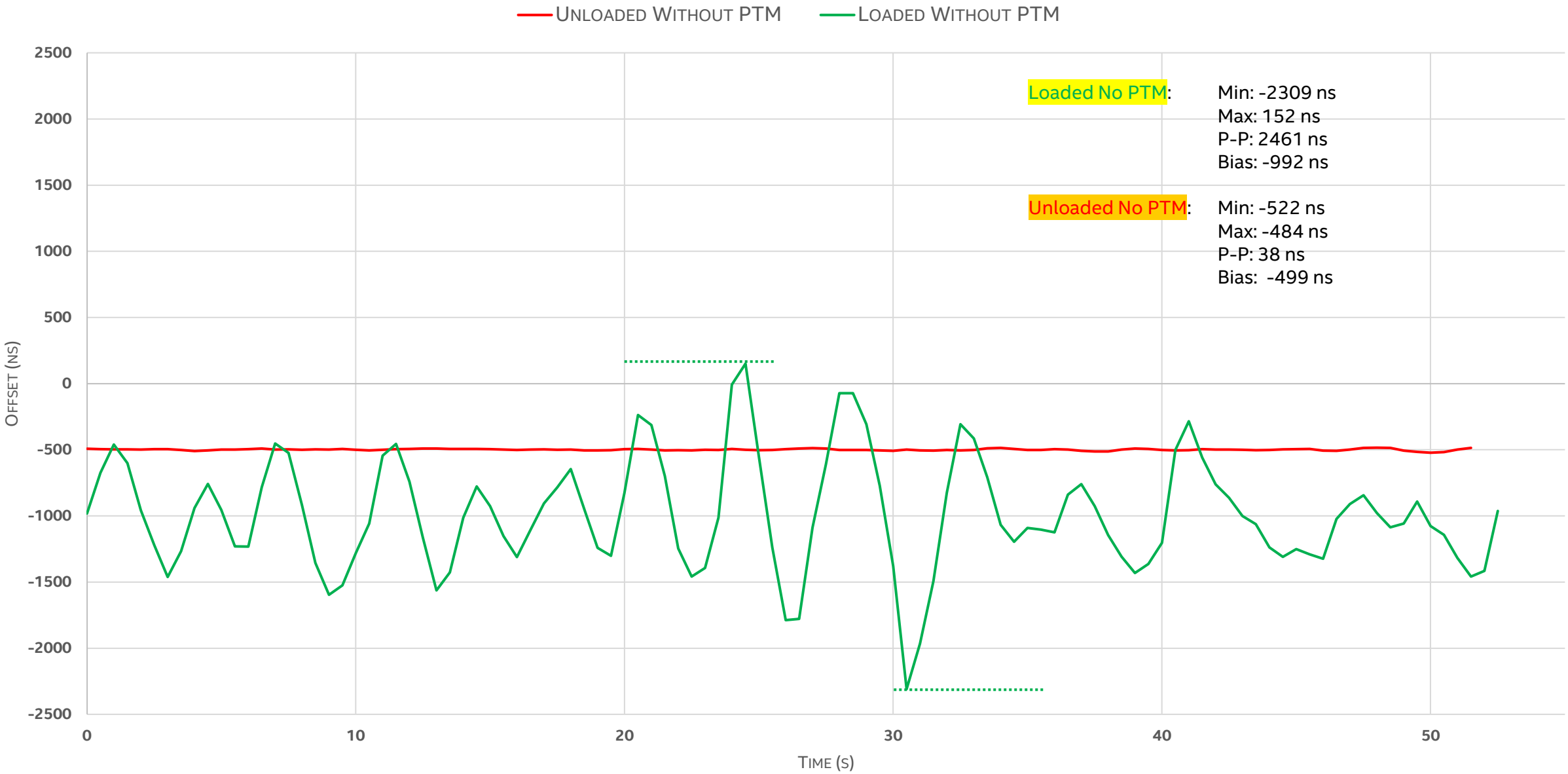
Test Setup: PTM Synchronization



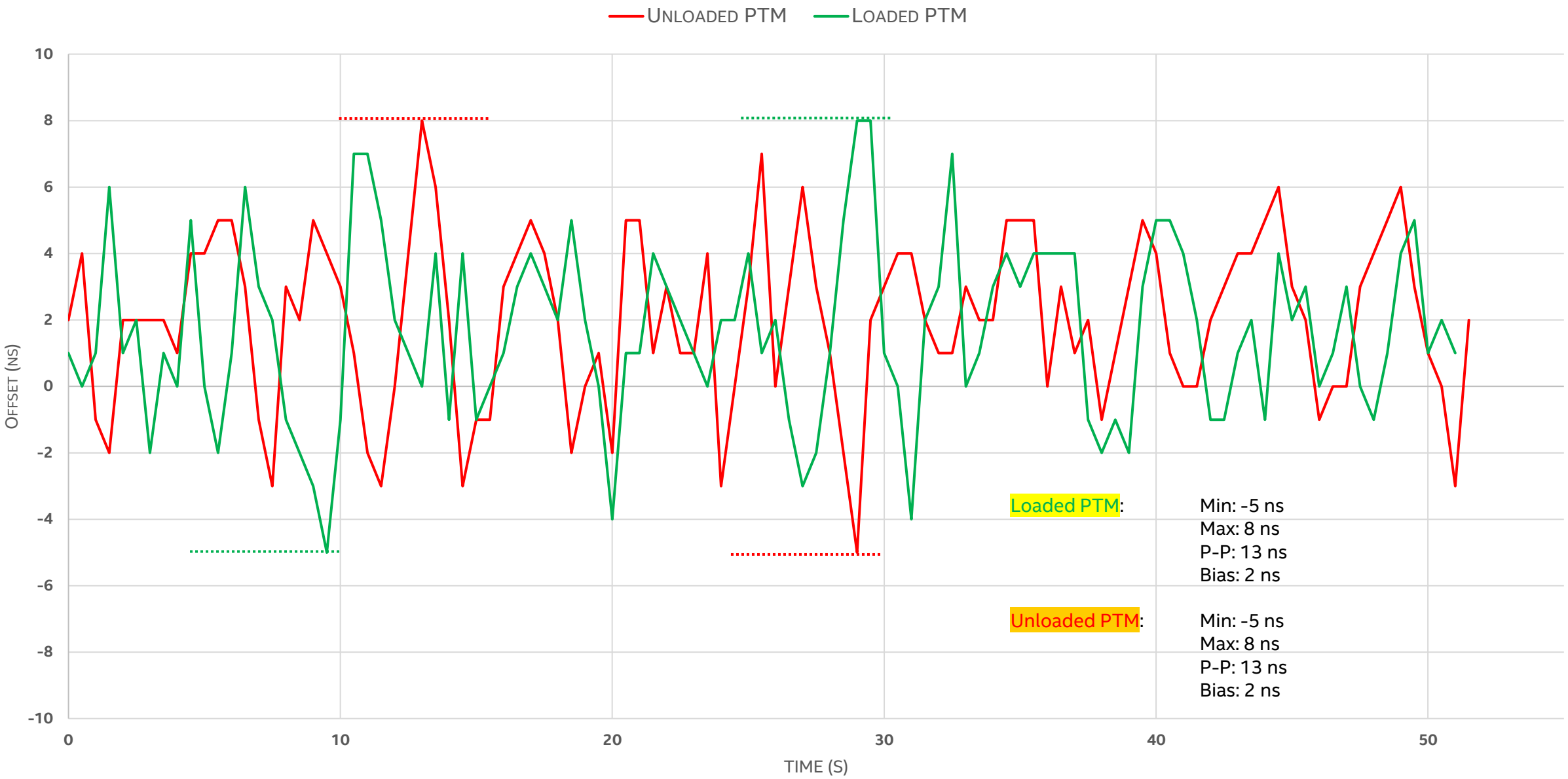
Measurements

- Software (No PTM) Offset
  - Idle System
  - Loaded System
- PTM Offset
  - Idle System
  - Loaded System

Software Offset Measurement: Loaded vs Unloaded System



PTM Offset Measurement Loaded vs Unloaded System



Measurement Takeaways

- The software method of offset computation introduces an unknown bias and significant jitter
  - PHC2SYS is not aware of the bias – due to “sampling error”
  - The bias varies based on system load
  - In the best case (idle) the offset jitter is more than double that of PTM
- PTM offers a solution that:
  - Does not introduce bias
  - Has lower jitter
  - Is insensitive to system load

Using PTM to Improve Application Performance

- PTM requires support in the endpoint device and root port
- Linux enables PTM support by default
- Requires driver support
- Linux PTP, by default, using PTM timestamps when they are available to discipline the system clock
- In the network stack, PTM cross-timestamps (offsets) are read using PTP_SYS_OFFSET_PRECISE targeting the associated PHC device

Next Steps and Call to Action

- Repeat using other loads
- Validate for longer “runs”
- Adopt methodology for measuring synchronization within the platform
- Agree on a standard CPU/IO load for measurement

Platform Configuration

- Intel Atom® x6427FE
- Linux kernel: 5.4.138
  - https://github.com/intel/linux-intel-lts
- Linux PTP v3.1+
  - http://linuxptp.sourceforge.net/


intel