# IEEE 1588 PTP



# The IEEE 1588 Standard

**The IEEE 1588 Standard**

Synchronizing time between multiple computers in the network has always been a challenge. What is, in the first step, a trivial requirement to equip two or more computers with the identical time is, on closer inspection, a very complex problem. The problem can be defined as follows:

**Distribution of the time of a very precise time source (atomic clock, GPS) on one or more computers in the network. Only the existing network infrastructure is available to distribute the time.**

To solve this problem, the NTP protocol was developed by David L. Mills at the University of Delaware in 1985 and published as RFC 958. With this protocol, it was now possible to assign several computers to the time of a time server. The accuracy is reached in the microsecond range (1x10-6 sec.) And was sufficient at that time.

As early as 1990, the T & M sector recognized the need to have significantly more precise systems. At this time, GPS systems were also available that could provide a much better accuracy, but this alternative was not attractively priced. In addition, the requirements for improved accuracy with the then existing infrastructure (should) needed to be achieved. For example, at the beginning of the nineties, a working group around John C. Eidson was founded at Hewlett-Packard to develop a concept for the improvement of the NTP standard. The results published in 2000 by John C. Eidson were so interesting for the T & M companies that they wanted to set up their own IEEE standard for this new type of time synchronization. In 2002, the IEEE 1588-2002 was completed and released. With the help of this standard, accuracies in the sub-microsecond range became possible. In 2008, the standard was once again adapted to the current requirements and defined as IEEE 1588-2008. This standard is valid still today. Currently, the IEEE Working Group is working on a new version of the standard, John C. Eidson and Doug Arnold, who are now co-chairmen, are responsible to meet the extended requirements of the standard. The following new requirements were:

- Improvement of time synchronization accuracy
- Applicability to localized systems with the possibility of use in larger systems
- Administration-free operation
- Suitability in high-end devices as well as in low-end devices
- Management control for redundant and fault-tolerant systems

Not only has the IEEE Working Group worked on the improvement of the standard but a merger of various institutions and companies, including the CERN Research Organization, the GSI Helmholtz Center for Heavy Ion Research and National Instruments, have further developed the standard for their requirements and published it under  the project name "White Rabbit" [3]. The objectives of the White Rabbit project were [4]:

- Accuracy of time synchronization in the sub-nanosecond range
- Thousands of nodes within the PTP domain
- possible node distances of more than 10 km
- Ethernet-based gigabit network with reliable data transmission
- Open source hardware, firmware and software
- Commercially available hardware from different manufacturers

The current version 2.0 of the White Rabbit standard was released in 2011 under the "CERN Open Hardware License". Currently, this standard is already used in various projects. CERN uses the standard for a control system of the "Large Hadron Collider" and for its neutrino telescope KM3Net in the Mediterranean Sea and the "GSI Helmholtz Center for Heavy Ion Research" uses it for the particle accelerator FAIR.

In order to understand the IEEE 1588 standard better, it is important to remember the problems of how such an exact time synchronization can be so complex. There are basically several problems:

1) Each clock within a computer is implemented as a counter. The clocks of a clock source are counted and the time is calculated via the frequency of the clock source. Since clock sources can never produce a constant clock, two parallel clock sources can never have the same clock. Thus, an adaptation of the clock frequency of the slave to the master is necessary.
2) The Ethernet used as a rule corresponds largely to the IEEE 802.3. Ethernet can unfortunately not provide a time-deterministic connection between two computers. Runtime may vary from one program to another. Thus the run times of the synchronization data can also vary, which can lead to errors in the synchronization.
3) The run times within the computers, ie in the drivers and the network stacks of the operating systems, are also not deterministic. Modern operating systems operate with "preemptive scheduling", which means that any process within the operating system can be interrupted and continued at a different time. Thus, processing times within the operating system can no longer be predicted deterministically. This blur leads again to errors in the time synchronization and must be taken.
4) A consumer product (network card, network chip) must be used as transmission media. The existing infrastructure (networks) must be used. Therefore, no independent hardware for the time synchronization can be used.

The IEEE-1588-2008 standard contains various aspects of synchronization and its environment. First, the devices in the networks (PTP domains) and their arrangement. Each of the devices or nodes in a PTP domain containing a PTP functionality are referred to as "clocks" and may occur in four.
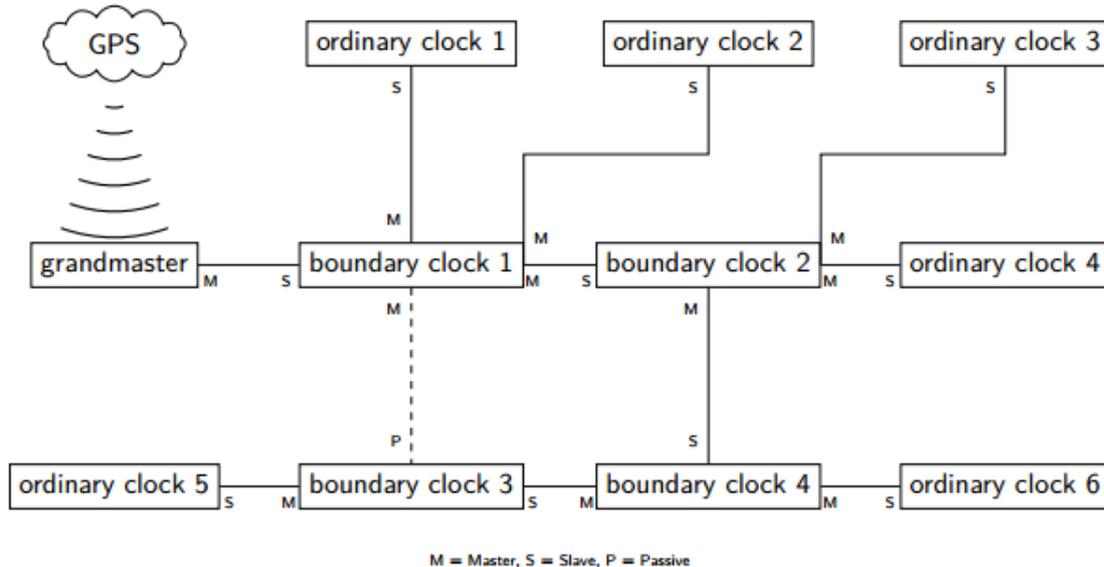


Figure 1: Schematic structure of a PTP network [2]

**Ordinary Clock:**
An "Ordinary Clock" is a simple clock (a port, usually a client) that is connected to a master as a slave and adjusts its time to it.

**Boundary Clock:**
A "Boundary Clock" is a clock which contains several "Ordinary Clocks" (several ports) and which can synchronize several slaves with their time as master. The "Boundary Clock" can also be connected as a slave to a master and its time can be adjusted.

**Transparent Clock:**
A "Transparent Clock" is a clock that does not actively intervene in the time synchronization; it is more of a hardware that provides the time synchronization data packets, typically a network switch. "Transparent Clocks" correct the time stamps within the data packets by the amount of time spent within the hardware.

**Grandmaster Clock:**
The Grandmaster Clock is an "Ordinary Clock" that normally has access to GPS or other very accurate time, providing this very accurate time for all subordinate nodes.

Within the PTP network, each clock (port) can act as master or slave. Within a PTP domain, however, there can always be only one master. The master actively distributes its time to the slaves. It is thus responsible for the synchronization of the slaves in its PTP domain. A slave is a passive element within PTP that responds to the

master's master synchronization requests. In principle, each clock (port) can function as master or slave within PTP. The decision as to whether the clock has to work as master or slave is determined when entering the PTP network. The "Best Master Clock" algorithm was defined in IEEE 1588 standard. This algorithm regulates the clock with the best accuracy, the role of the master.

Synchronization of the time between master and slave is carried out via so-called "event messages". These messages are distributed by the master via the UDP network protocol in the network and answered by slaves. There are two different methods for synchronization, the one-step synchronization, and the two-step method. The difference in the two methods lies in the number of messages which must be exchanged and can be seen in the following diagram. The two-step method sends four messages to perform a synchronization step. The one-step method requires only three messages, but this is only possible with very good hardware support and cannot be found in any PTP domain.

In order for a slave to perform the synchronization, only four time stamps (t1, t2, t3, t4) are necessary. The time stamps are generated when sending or receiving certain messages. The slave receives these four time stamps by exchanging the following "Messages":

• SyncMessage, the master allows the slaves to start a synchronization; in the "one-step" method, the time stamp t1 is contained in this message
• Follow_Up Message, this message is analogous to the SyncMessage, whereby now the time stamp t1 is sent, if the "Two-Step" methods are applied
• Delay_Req message, the slave responds to a Sync or Follow_Up message and asks the master for another timestamp
 • Delay_Resp Message, the master uses this message to send the last time stamp t4 back to the slave

The following diagram shows the exact flow of the PTP communication and the associated exchange of the required time stamps. The time stamps t1 and t4 are recorded on the master side and are sent to the slaves with the messages. For this node to have its four time stamps, t2 and t3 are created on the slave side upon receipt of the master messages.
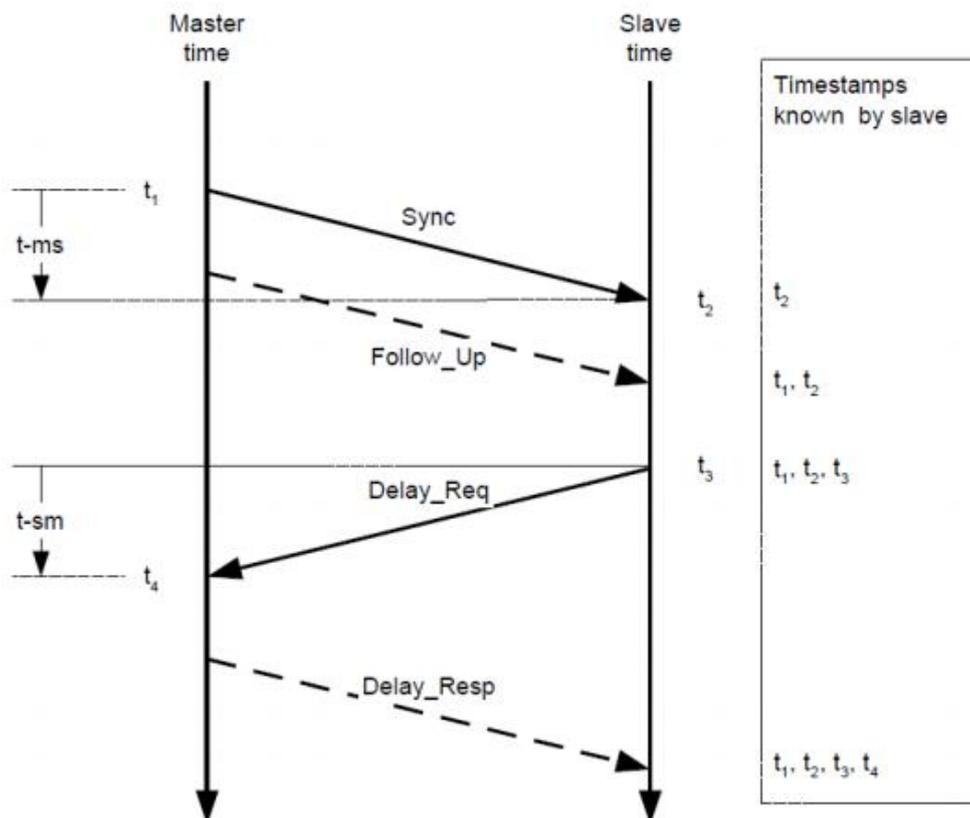
Fig. 2: Schematic sequence of the messages [1]

With the help of the times t1, t2, t3, and t4 the "meanPathDelay" can be calculated after the exchange of the messages. The "meanPathDelay" specifies how long a message used for the transmission in the network, ie the delay. As can be seen from the formula below, IEEE 1588 assumes that the latency is identical in the network for sending and receiving.

$$\langle meanPathDelay \rangle = \frac{(t2 - t1) + (t4 - t3)}{2} = \frac{t_{ms} + t_{sm}}{2}$$

Figure 3: Formula for calculating the meanPathDelay [2]

Thus, the "meanPathDelay" is used for the runtime correction (in the network) for the calculation of the own time offset.

- If you look at the diagram above, you can conclude that the closer the time stamp tn is set or determined at the hardware interface, the more accurate the calculated "meanPathDelay" is. This observation then leads very quickly to the realization that part of the IEEE 1588 functionality should be accommodated optimally in the corresponding network chips. 2012 Intel then presented their network chips of the Intel series I21x which contained for the first time a support for IEEE 1588. With the presentation of these network chips, the time

had now come to think about commercial solutions in the T & M area. With this consumer chip set, Intel had for the first time created the possibility to use the IEEE 1588 for a wide range of devices. These network chips contain all time-critical parts of the IEEE 1588 Standard:

- Hardware clock: An internal counter and the necessary actuators for the adjustments of the counter (frequency adjustment)
- Hardware Timestamping: The timestamping of the sending UDP PTP data packets
- Hardware Timestamping Grabber: Capture the timestamps when receiving UDP PTP data packets
- Hardware Based Time Events: Generating timed triggers

On Linux there are already some implementations of different quality. For the Windows operating system, however, there are no drivers from Microsoft or Intel to support this functionality of the Intel I21x network chips. Although some commercial implementations are available, they are only of limited value in terms of price.

With the help of the Intel I21x network chips, excellent time synchronization in double-digit nanoseconds can be achieved with little effort, which can also be maintained for a long time.
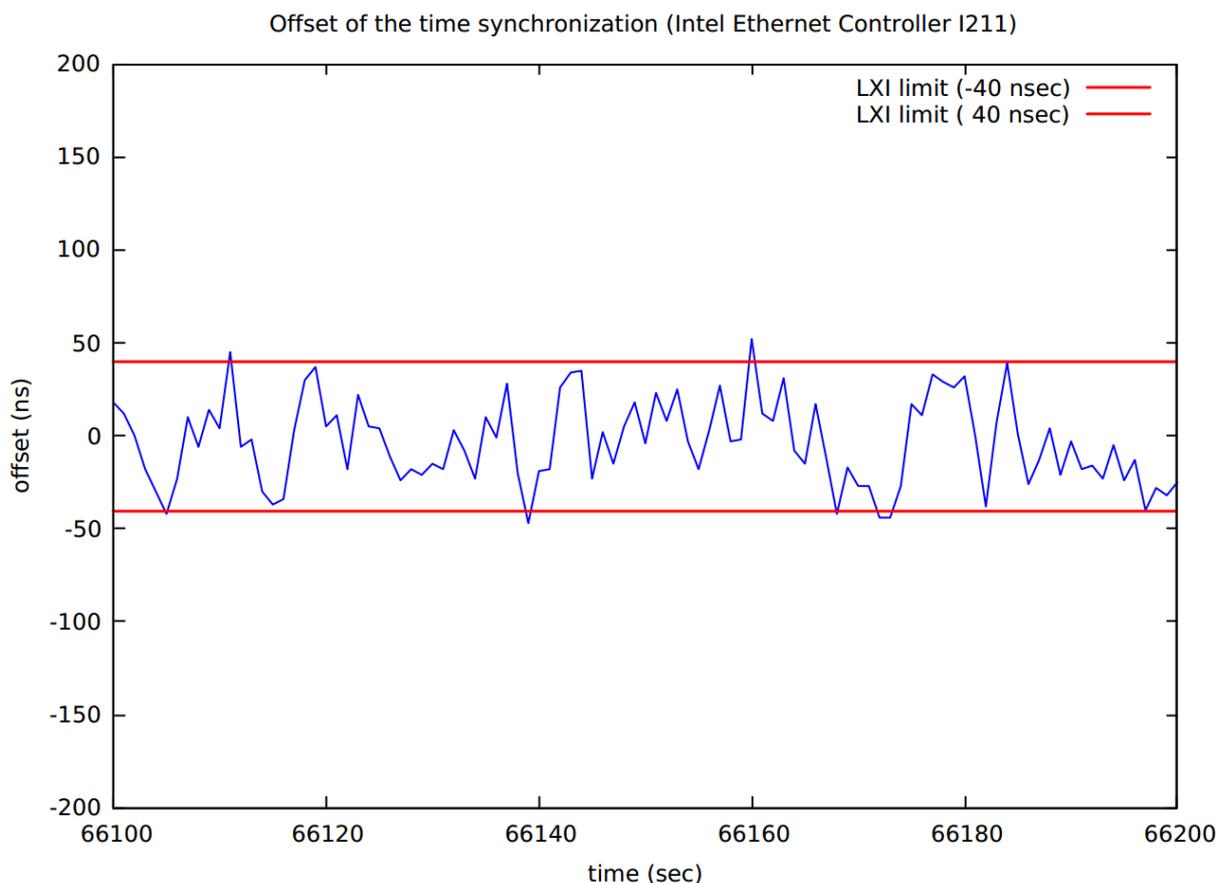


Figure 4: Measurement of the slave offset over 100 seconds [2]

As the diagram above shows, the offset between the master and the slave always fluctuates around the zero point. Due to the calculated offset, the current time of the slave must be adapted. As is easy to see, the time cannot simply be corrected, ie, the determined offset can be directly applied, since the time would then jump into the past or future. A more sensible approach is to manipulate the clock of the counter in the slave. The Intel I21x chips have a special register with which the frequency of the counter can be adjusted indirectly. The Intel I21x chip updates its local clock by 8ns every 8 ns, and an offset of n x 2-31ns can be defined, which is added to each of these updates. This offset is signed and thus can be used to reduce or enlarge the counter frequency. Due to the offset to the master, the corresponding error in the frequency and the necessary frequency adjustment can now be determined. The underlying frequency adjustment algorithm is responsible for the stability of the watch.

Within the network, interference can also occur which ultimately affect the stability of the clock. The diagram below shows disturbances caused by network collisions and the resulting transmission repeats in the Ethernet.
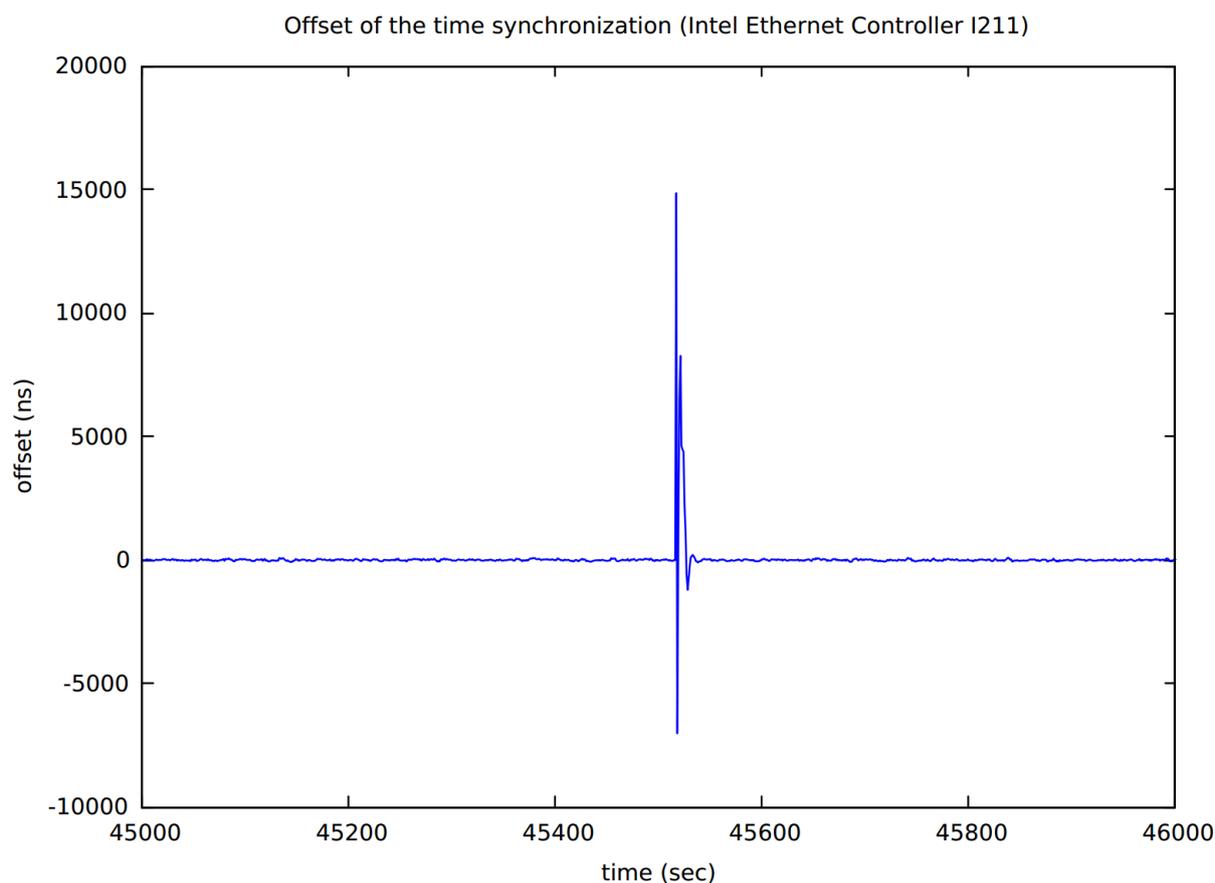


Figure 5: Measurement of the slave offset over 100 seconds [2]

In order to be able to recognize and evaluate such disturbance variables, a Kalman filter can be used, for example.

In the T & M area, with the help of the exact time synchronization via IEEE 1588, many new and interesting approaches are now emerging.
If one considers that a trigger signal requires approx. 5ns running time per meter, one can certainly imagine some measurement tasks which can be solved better and more precisely with the help of the IEEE 1588 time synchronization. For measuring tasks, which have to do with satellite signals, for example, it can happen that measuring devices are 100 and more meters apart, the approach with IEEE 1588 would certainly be an alternative.

Also in the course of the increasing distribution of the IoT sensors and the offline measurement data acquisition, an accurate and stable time synchronization of the individual measuring sensors would be of great advantage. Thus, the individual IoT sensors can continuously supply their data and store it in a cloud, for example. This data can then be simply correlated with an offline measurement data evaluation via the time stamp, which was created during the measurements.

Since IEEE 1588 also defines time-controlled events, for example, several devices can process sequences of measurement tasks without having to be connected with trigger lines or the like. For example, a signal generator could process its frequency list in a timed manner and the analyzer can perform its measurements. Both tasks would then be controlled by the time-bound.

With the use of IEEE 1588, a wide range of measurement tasks can certainly be implemented much more easily in the field of mobile communications, since the temporal coupling between base station and mobile radio is of relevance for various measuring tasks.

These application examples are all very promising in the first approach, but are based on the assumption that the manufacturers of the measuring devices have also implemented this standard in their devices. Even if all parties would use the IEEE 1588 standard, a further standardization is still necessary to coordinate the individual actions and tasks.

In order to make this easier for instrument manufacturers, the LXI standard (Lan eXentsion for Instruments) was launched in 2004 by the leading measuring instrument manufacturer. In 2008, the IEEE 1588 standard was included in the LXI standard and extended by some extensions, such as the Lan Event Messages. With the help of this standard it is now possible for measuring instrument manufacturers to efficiently use the advantages of IEEE 1588 in measuring instruments. Until the introduction of the Intel I21x network chips, cost-intensive or proprietary hardware

was always required to integrate the IEEE 1588 standard in the measuring instruments. With the Intel network chips, it is now possible for the first time to realize this standard with consumer electronics.

As early as 2014, the LXI consortium decided to provide a reference design for your members to further spread the use of the standard. After the core functionality of the standard was available in the LXI reference design 2016, the LXI consortium, together with the German company TSEP, began evaluating the possibility of a reference implementation of the LXI Time Synchronization (corresponding to IEEE 1588) and to create a prototype. In February 2017, the first prototype was presented by TSEP at one of the regular meetings of the LXI consortium in the USA. Both on Linux and under Windows, the prototype was running on a commercially available computer board with Intel I211 chip.

The LinuxPacket "LinuxPTP" was used for the Linux prototype. However, the implementation of the actual clock functionality was not sufficient to meet the required framework conditions of the LXI and IEEE 1588 standards. TSEP has therefore reimplemented this. The control algorithm for frequency tracking of the IEEE 1588 Clock has been completely redesigned and optimized.

As mentioned above, Windows does not provide any support for IEEE 1588, so the complete clock functionality as well as the driver support of TSEP was created. TSEP has anchored the necessary drivers for the control of the Intel I21x network chips for the timestamping and the recognition of IEEEE 1588 packets in the Windows Network Stack. Both Windows 7 and Windows 10 tested this functionality.

The first discussions with the LXI Consortium on the provision of LXI Time Synchronization (equivalent to IEEE 1588) were consistently positive and are likely to lead to a free provision for LXI members this year. TSEP also plans to provide its implementation for other non-LXI members.

In summary, it can be said that with the commercial use of Intel I21x network chips it is now possible to use IEEE 1588 in T & M devices. The new solutions developed with IEEE 1588 extend the possibility to solve measurement tasks in the T & M area by a new dimension. If additional extensions like the LXI standard are offered in T & M devices, complex measurement tasks can be structured more simply and efficiently. Self-configuring systems are now possible with these approaches, since device data and actions can be configured, exchanged and triggered network-wide. It remains to be seen how the measuring instrument manufacturers react to this new technology, but there is a great potential in it.

**Bibliography:**

[1] IEEE standard for a precision clock synchronization protocol for networked measurement and control systems, 2008.

[2] Bachelor Arbeit von Dominik Richstein, „Usability of the Intel Ethernet Controllers I210, I211 and I350 for the LXI Extended Function Clock Synchronization"

[3] CERN. White rabbit specification: Draft for comments, 06.07.2011.

[4] White Rabbit Project. White rabbit wiki, 2016.

**Authors:**

**Peter Plazotta, Dipl. Ing (FH)**
CEO of TSEP, and has been involved in the design and development of system software for the past 30 years in the fields of T & M, telecommunications and automotive.
Peter.Plazotta@tsep.com

**Dominik Richstein, B. Sc,**
Studied at the Technical University of Ingolstadt and has written his Bachelor thesis on the topic of IEEE 1588 and Intel I21x network chips at TSEP.
Dominik.Richstein@tsep.com