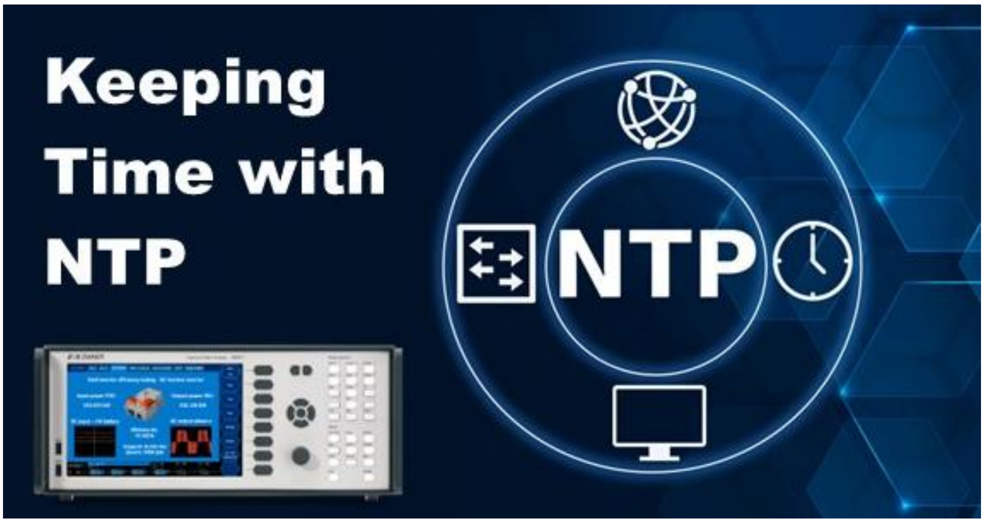

Whitepaper

V1.0

PRECISION IN POWER – THIS IS OUR GUIDELINE!

NTP Time Synchronization for Measuring Instruments

At the beginning of this whitepaper, you will be introduced which role time and clocks generally play in the power analyzers of the LMG series.
If you think about a measurement you might do, the measurement data will always have a time axis. The measurement data are timestamped each time data are acquired. The timestamps may originate from a simple *timer* which starts at zero. With this method you could compare only values within one measurement to each other because the measurement data shows only *relative* timestamps. But if you want to compare data from different measuring situations, from instruments, or locations, this relative timestamp is not very helpful. Therefore, the time-base of the LMG is implemented as a *clock.* This clock is used to assign *absolute* timestamps to each measured quantity.

But what is actually time *synchronization*? Time synchronization, or 'clock synchronization' describes two clocks ticking at exactly the same speed and showing the same absolute time. You might know that from setting your wristwatch by adjusting it manually to another clock, the 'reference clock'. If you do this periodically every noon, we could speak of a kind of synchronization.

As described above, the clock inside the LMG is used to timestamp the measured quantities. This clock might be set by manually entering the time using the 'eyeball and wristwatch' method. As you could imagine, even if you put great effort into setting the time as exactly as possible, you will never reach an accuracy lower than 1/10 of a second. But the measured quantities can be updated up to 100 times per second (and the raw sample values even faster). So, to set the clock within this rough accuracy range might not be sufficient. Thus, we offer automatic time synchronization support for the LMG.

There are 3 major scenarios in which a good time synchronization can be helpful:

- You have a parallel measurement at different locations which do not allow having any kind of cable connection between them. To compare (or correlate) the data from one location with another location, you need absolute timestamps.
- You have a parallel measurement at one location, but with different instruments from different manufacturers. These instruments usually do not have compatible interfaces, so you need absolute timestamps from all involved instruments to correlate the data.

What is NTP and where do I get it?

The network time protocol (NTP) is a service which is available in TCP/IP networks. In principle, there must be an NTP server that distributes the absolute time, and there are NTP clients (in our case, measuring instruments), which synchronize their internal clocks to the clock of the NTP server.

There are several possible options for NTP servers:

- Many National Metrological Institutes (NMIs) offer a NTP service, which is based on their internal reference clocks. To connect to them, you need, of course, access to the internet.
- There are special time receivers available which get their time from GPS, DCF77 or similar broadcast systems. These devices can serve as local NTP server which don't need a connection to the internet. They can provide a local network with very accurate timestamps.
- In the simplest case any computer in a network can offer the NTP service. In this case, the *absolute* accuracy of the timestamps might be worse than above, but as long as *all* involved measuring instruments get the same wrong value, this doesn't matter.

What types of errors do clocks have?

To understand how NTP disciplines a wrong ticking clock, you first have to understand the two major errors, a clock might have. At first a clock may have an offset error. An offset error of a clock could be fixed by ‘setting the time correctly’. The offset error is a long-running, *constant* deviation of a clock compared to a reference clock. For instance, by manually setting the clock in the LMG, it would be impossible to eliminate an offset error smaller than maybe 0.1 s, due to human and system reaction times.

The next error we have to go over is the clock drift. The clock drift specifies the increasing or decreasing of the *offset error* over a long time period. The clock drift is mainly caused by the oscillator supplying the clock with a (ideally) constant frequency running too fast or too slow. In older mechanical clocks, this error was caused by mechanical wear-out; in modern clocks like the one in the LMG, it is mainly caused by fabrication tolerances or thermal drift of the crystal oscillator.

How does NTP determine the correct time?

To eliminate the errors mentioned in the previous section, NTP has to determine the current time fetched from a NTP server first. If NTP continuously ‘knows’ the correct time it can continually calculate and correct the offset error and the clock drift of the instrument's oscillator. But, unfortunately, the IP packet transfer from a server to a client does not proceed with zero delay. Even when having a good connection to a server over the internet, you might encounter delays about several milliseconds or more. In local area networks these delays are typically lower, but they will never be zero. So, receiving a single timestamp from an NTP server always entails an unknown offset at the receivers end which may significantly decrease accuracy.

To remove these unknown network delays when exchanging timestamps, NTP uses the so-called ‘On-Wire-Protocol’. This protocol operates by exchanging at least three timestamps and adding a fourth timestamp after the receipt of the response by the server. By using these four timestamps the NTP client is able to measure the network delay between the server and itself and can determine the correct absolute time by using these timestamps.

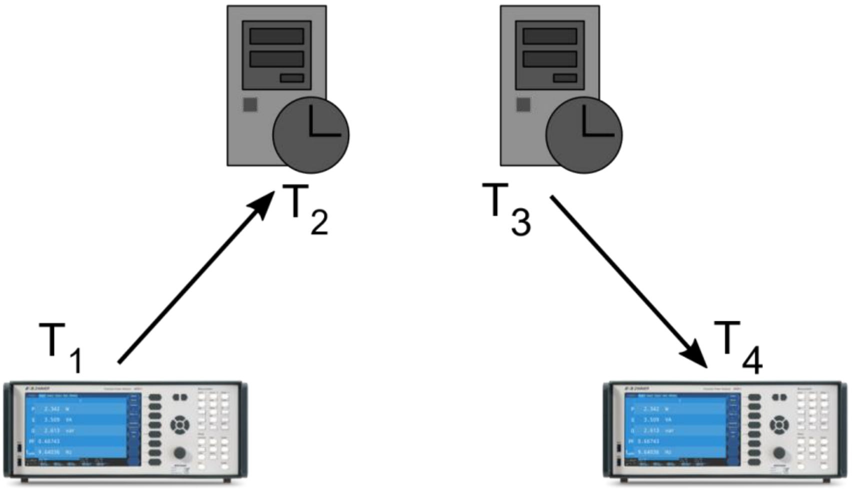

Figure 1: NTP Time Determination Process (On-Wire-Protocol)

As indicated in Figure 1, there is an NTP packet sent from the client (LMG) at T1 to the server, received at T2. The same server sends it back at T3 to the same client, receiving at T4. Timestamps 1 and 4 are generated on the client side with its local clock, timestamps 2 and 3 are generated on the server side with its local clock. After this process, the time offset between the client and the server clock is calculated:

$$\theta = \frac{(T_2 - T_1) - (T_3 - T_4)}{2}$$

This offset is statistically evaluated to remove deviations which will always happen during such a procedure. This method assumes, that the network delay between the server and the client is symmetric and equally distributed on both directions. Having an asymmetric delay in your network, NTP will always generate an error when determining the clock offset.

In all cases, the correction of the delay will not be ideal, and a small uncertainty will remain. For NTP, literature specifies a typical uncertainty about 10 ms for synchronization over the internet and about 200 µs for local area networks.

When the client has finally determined its offset to the server's clock, it must discipline its own system clock to run at the same speed as the servers' clock to continuously keep both clocks synchronized. For doing so, NTP uses a *Variable-Frequency-Oscillator* (VFO) which can be sped up or slowed down. For offsets smaller than 128 ms NTP adjusts the time by changing the oscillator's frequency. For offsets greater than this value, NTP does a bigger time step to speed up the synchronization process.

NTP describes both, the protocol and the corresponding mechanisms for correcting the system clock. But it isn't quite as simple as described above. There are many complex algorithms for filtering and other stuff far beyond the scope of this whitepaper. More details might be investigated e.g. through David L. Mills book "*Computer Network and Time Synchronization*", where Mills, the inventor of NTP, describes all mechanisms used and the whole protocol in detail. This section of the Whitepaper was intended to give a really quick overview on the basic operations of NTP and how the time determination process works.

NTP configuration parameters

As NTP is a highly configurable process, most configuration of the NTP implementation in our instruments was set to deliver as accurate time synchronization as possible and was determined through real use case experiments. At least there are two parameters, called *minpoll* and *maxpoll* that should always be accessible by the user and have to be adjusted to fit the environment NTP is used in to fulfill individual demands.

The *minpoll* and *maxpoll* values configure the polling interval. Polling refers to the process where the NTP client sends a NTP packet to a server and receives an answer as depicted in Figure 1. Those two parameters define the interval range NTP may choose for doing its polling. The interval is expressed in powers of two in seconds with a minimum of 4 (equal to 2^4 s = 16 s) and maximum of 17 (about 2^{17} s ~ 36 h). If you are setting these parameters to different values, NTP determines the best polling interval within the specified range. It tries to raise the interval after some time when the system oscillator has stabilized to minimize network traffic. A disadvantage of this NTP behavior is the loss of accuracy with higher polling intervals because the system clock can drift away further in longer polling intervals. To avoid this, it's possible to set *min-* and *maxpoll* to the same values to force a fixed polling interval. The impact of those parameters on the accuracy of the instrument's clock will be described more in detail in the following sections.

For synchronizing with local timeservers which are under local control a fixed value of *minpoll=4* and *maxpoll=4* is recommended to achieve the best synchronization results. For synchronizing with most public servers *minpoll=4* and at least *maxpoll=6* or greater are best practice values to minimize the risk of getting banned from these servers by exceeding their rate-limits.

NTP performance parameters

If you synchronize your instrument to an NTP server, there are different methods to estimate how good the synchronization actually is. It is important to keep in mind that it's only an estimation.

NTP has many internal values and statistics where it is necessary to choose the correct ones to make a good estimation. The best one for that is the 'estimated error', sometimes called 'system jitter'. Simplified, the estimated error is calculated using the jitter of the clock's offsets calculated after each polling operation. In a nutshell, the smaller the estimated error the better the synchronization of the LMG with the NTP server. It is an estimated value which shows if the synchronization improves or diminishes. It is a best practice to observe this value over a period of time in a specific environment to determine the smallest value reachable in this setup. If then the estimated error increases or decreases you know whether the synchronization quality increases or decreases.

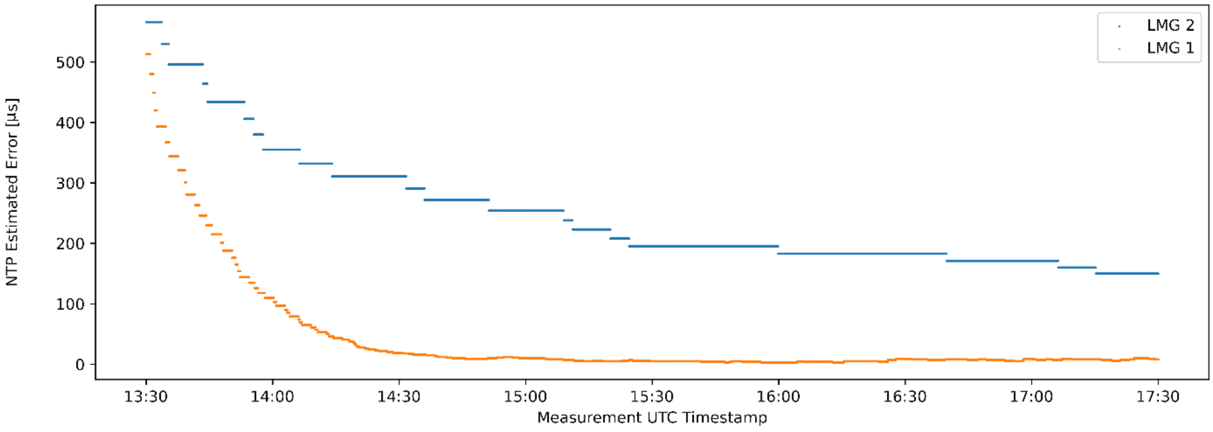

Figure 2: Observing the estimated error over a time span of four hours

For this figure, which shows the value of the estimated error, two instruments of LMG series were chosen. The different behavior of LMG1 and LMG2 originates from the different polling settings. LMG1 was set up with *minpoll*=4 and *maxpoll*=4, whereas LMG2 was set up with *minpoll*=4 and *maxpoll*=6. As expected, shorter polling intervals result in a smaller estimated error.

NTP uncertainity experiments

In the preceding section the estimated error was discussed. But this parameter is not a metric for the absolute uncertainty. The uncertainty can only be determined by putting a reference clock near the instrument and by comparing the exact time when the seconds in both devices increase.

To examine the uncertainty of a NTP synchronized instrument clock we used the pulse per second (PPS) signal generated by our local time-server to compare the moment when the second changes in the server against the instrument. This way it is possible to measure the absolute offset of the clock within the LMG against the server and other instruments as well. By using a PPS signal, it’s impossible to measure offsets greater than one second. Because ‘PPS’ means Pulse Per Second, it is nothing different than a pulse occurring at a frequency of 1Hz. This pulse does not contain any other time information than the exact moment the start of the second occurs.

In the experiments described below, our main concern was focused on the clock offset within two identical LMGs and not on the instrument’s offsets against the reference clock. This is very important, because the absolute deviation to the reference clock is not as meaningful as this value. If the difference between two instruments is known and stable, this difference can be taken into account, when evaluating the data.

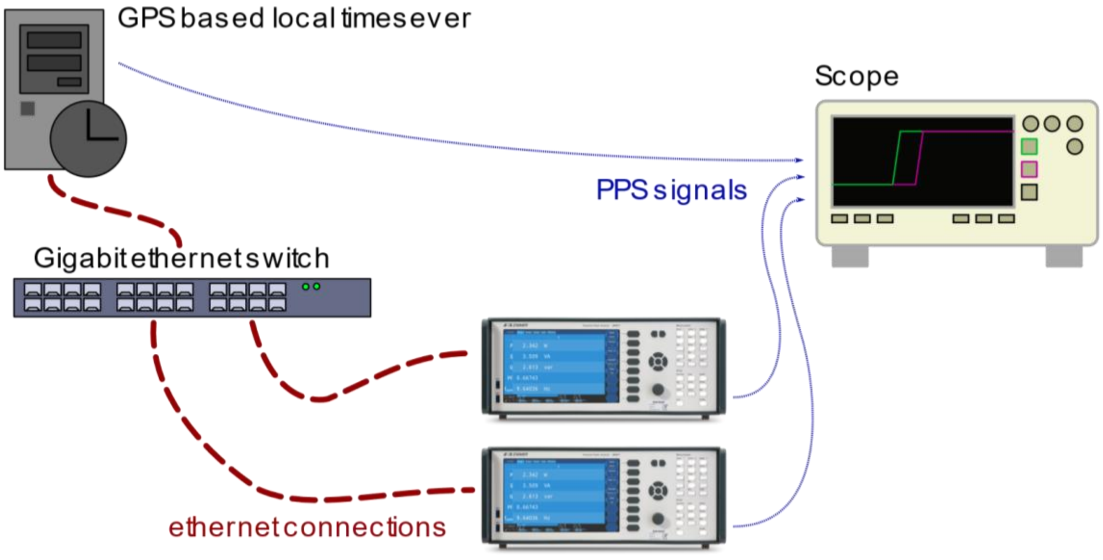

Figure 3: NTP accuracy investigation

Figure 3. shows the setup which was used to create the following table containing the average clock offsets. Figure 4. outlines the measurement done with the scope. Δ_{ref} in the figure outlines the absolute offset of LMG1 against the reference clock. This absolute measurement was not used in the further process. The important value is Δ_{lmg} which describes the offset between the clocks of the two LMGs used.

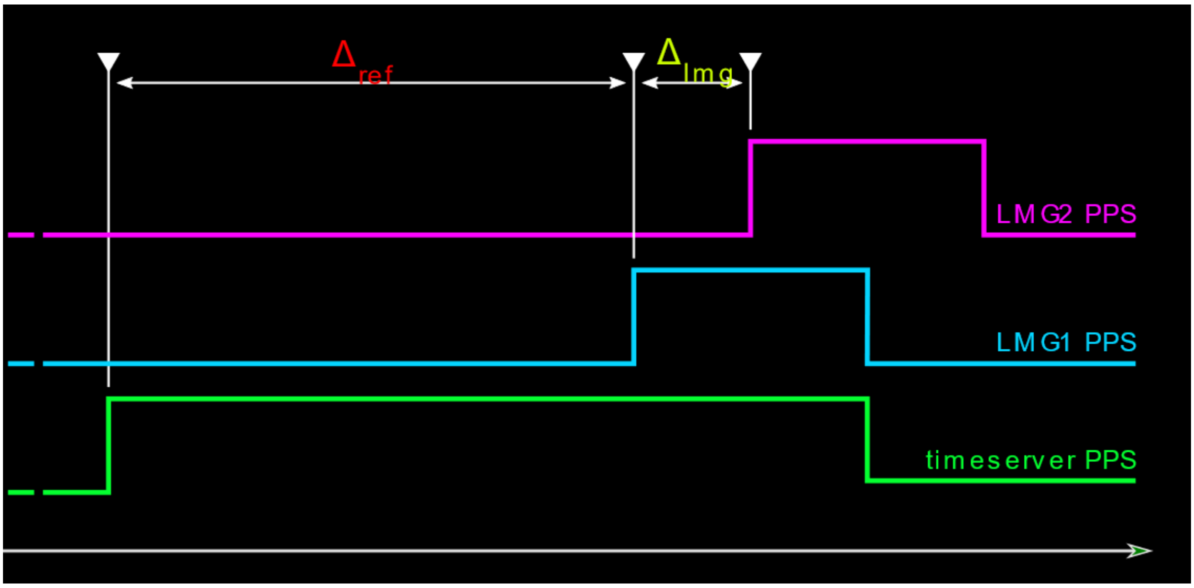

Figure 4: PPS comparison on scope

The following table summarizes the average clock offset $\Delta_{lmg_{avg}}$ of the Δ_{lmg} between those two LMGs averaged over the time of Δt.

Server	Δt	$\Delta_{lmg_{avg}}$ / ns
Local NTP Server	3h	500
debian.pool.ntp.org	10h	16 500
	3h	54 700
ptbtime3.ptb.de	3h	172 800
Local NTP server	1h	333 500
ptbtime3.ptb.de		
ptbtime1.ptb.de	3h	1 549 200
au.pool.ntp.org		

To interpret these values listed in the table, you should know more about the used servers. Those servers were chosen to provide different initial conditions like geographical distances from very short to typical and to very far away, and extensive network delay.

Local NTP server

The local NTP Server used in the experiments is a small 19” rack-mount timeserver bought from a German company. It has an integrated crystal oscillator with an uncertainty smaller than 0.03 ppm and uses a GPS receiver with an outdoor GPS antenna as time synchronization source. The timeserver was connected via an Ethernet cable to the local area network. The cable was plugged into the same switch where also both LMGs were plugged in. The timeserver was configured to be on the same subnet to assure ideal network conditions and to minimize the network delay between the LMGs and the server. The setup is depicted in Figure 3.

ptbtime1.ptb.de and ptbtime3.ptb.de

These are public timeservers of the Physikalisch-Technische Bundesanstalt (PTB), the national metrological institute of Germany. The PTB is also known for controlling the DC77 time signal radio station. It uses the NTP server version 4.2 and synchronizes their timeservers independent of GPS with their own atomic clocks.

debian.pool.ntp.org and au.pool.ntp.org

NTP pools are lists of NTP servers from which the NTP Daemon can choose the one which promises the most accurate synchronization. The first pool used here is the default pool preconfigured in the NTP configuration delivered to Debian GNU/Linux systems. The second one contains all public NTP severs from Australia that are on the Australian pool list. This pool was chosen to enforce the synchronization with a geographically very distant NTP server to Germany.

If you want to learn more about the NTP pool project just visit https://www.ntppool.org/. There is a list with all available pools grouped by continents.

NTP accuracy characteristics in detail

This section describes the accuracy characteristics shown in the table in the previous section in more detail and the reasons why they occur.

The time-server in the local area network offers the best accuracy. Best, the network path to the timeserver is as short as possible. Then the clocks of the LMGs could be synchronized in orders of few hundred nanoseconds on average under good circumstances between two LMGs. This is about 3 orders smaller than the value specified in literature!

The next scenario synchronizes the LMGs to the Debian NTP pool. Here, the NTP daemon can choose from several available servers from the pool. NTP automatically tracks the characteristics from a couple of servers and selects the best one of them. It can be seen that the average offset between the LMGs clocks decreases over a longer period of time. This happened because NTP reached higher accuracy after some time of operation by choosing the best NTP servers from the pool.

Next was the synchronization with the PTB NTP server. As you can see the average deviation is worse than synchronizing with the pool. The most presumable explanation is that by using the pool, NTP chooses an NTP server which has a lower network delay and which isn’t used by so many clients than the PTB server. It is reasonable that the well known PTB NTP server is under heavy use and might have additional network delay because of its utilization. Nevertheless, the uncertainty is well below 10 ms.

The following experiment consisted of synchronizing one LMG to the local timeserver and the other one to ptbtime3.ptb.de. As expected, the result is much worse than in the previous experiment shows why the use of different NTP servers with completely different characteristics and reference clocks is *not recommended*.

At least we can summarize that over internet NTP servers the clocks could be synchronized in orders of three- or two-digit microseconds between two LMGs in the same network depending on the chosen server and the quality of the network connection. In individual setups the results might be significantly different. It should be mentioned that asymmetric connections could impact the synchronization extremely.

The last experiment was made to enforce really bad circumstances by connecting one LMG to ptbtime1.ptb.de and the other one to any NTP server from an Australian pool to enforce a very high geographical distance of the servers. As you can see the difference of the clocks of the two LMGs increases to about 1 ms. Despite this is well below 10 ms, such a configuration is not recommended.

Compensation of thermal side effects

Besides the time synchronization itself, NTP has another great feature you could profit from when using it in an LMG. NTP stabilizes the system clock and thereby simultaneously also corrects a possible thermal drift of the crystal oscillator. When, for example the temperature in the room where the LMG is operating decreases suddenly, maybe because you open a window, the LMGs oscillator cools down and might run a little faster or slower. As mentioned before, this is called clock drift and NTP was designed to correct the clock drift. After an inherent delay, NTP speeds up or slows down the clock to restore the correct ticking rate to match again the clock of the NTP server.

The following Figure 2 illustrates the behavior of the clock with and without NTP enabled. On the vertical axis it shows the absolute offset in nanoseconds against the reference PPS signal. The horizontal axis holds the timeline of the experiment and shows the timestamps in UTC. At roundabout 07:45 the both LMGs were cooled down by opening a window and NTP was disabled on LMG1. The clock of LMG1 drifts away and produces an increasing offset over the time.

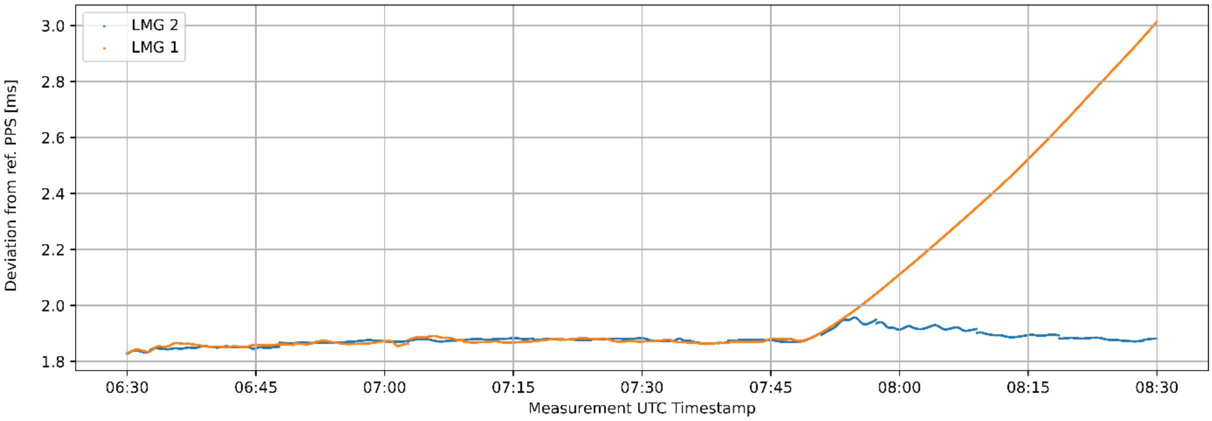

Figure 2: Reaction of the LMG Clock to Temperature Changes (With NTP Enabled (LMG2) and Disabled (LMG1))

By considering the graph produced by LMG2 it strikes that after a short delay the clock is disciplined to find its way back to the constant offset the clock had before. Compared to the behavior without NTP, the distortion produced through the temperature change is extremely minimized. The drift produced due to the temperature change on LMG1 could be approximated to ~0.26 ppm.

During another experiment, LMG1 ran with NTP enabled under a constant temperature environment and on LMG2 NTP was disabled and the device was cooled down with coolant spray. Later the temperature was raised to the original level. This happened when the curve of LMG2 suddenly changes its direction and decreases. Then NTP was re-enabled the graph snaps back to the correct constant offset. This process is depicted in Figure 5. As a reference, LMG1 ran with NTP enabled and without any temperature distortions.

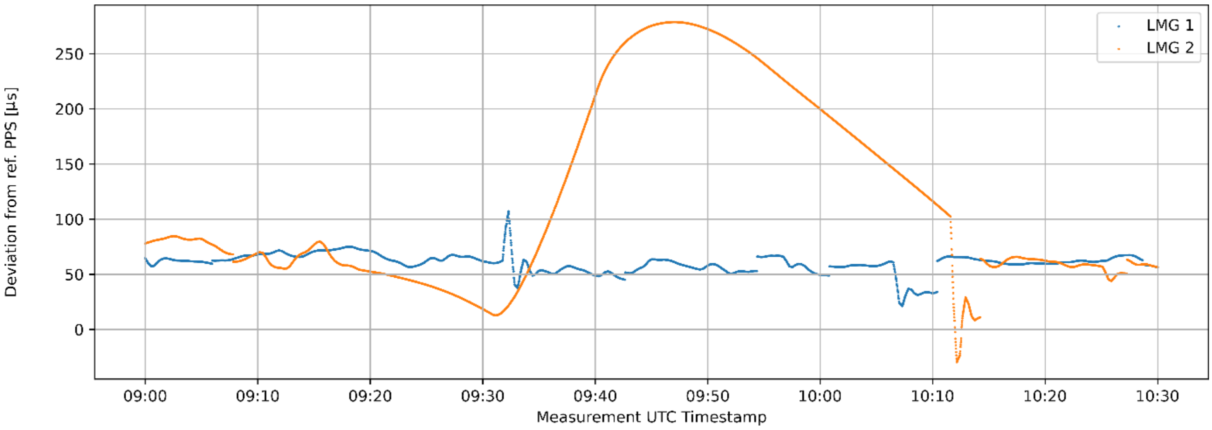

Figure 5: Reaction of the LMGs Clock to Temperature Rise and Fall Without NTP Enabled

NTP accuracy characteristics in detail

NTP is a great option to use if you need precise timestamped measurement data from one or more LMGs. An application might be a measurement over a longer time period where some events occur sporadically, and you need the timestamp of these events to compare to other measurement data. In this scenario you have to keep in mind, that the offset error of the clock may be greater compared to the reference clock or other devices than between two LMGs synchronized to the same NTP server.

The next useful application are longtime measurements in general. NTP helps here to compensate the clock drift. Without NTP enabled, the clock built into the LMGs could drift by about 4 seconds per day. NTP would keep this clock always synced, independent of the time passed on the condition of a stable connection to an NTP server.

The last reasonable application consists of one or more measurements done at distant locations and the measurement data needs to be compared afterward. When comparing, you must keep in mind that the cycles itself are not synchronized, only the clocks within the LMGs which put the timestamp onto the cycle data. If you utilize this scenario, you must set the cycle time of the LMGs to the same value. This issue is better described in the next section.

What you cannot achieve with NTP

NTP is neither a synchronization method to synchronize the sample values of the channels within different LMGs, nor does it synchronize the measuring cycles of different LMGs. So, within a longer time interval you might have a different number of sample values or measurement cycles in two different instruments.

NTP Startup

This section covers the first NTP startup. If you enable NTP on your LMG for the first time, you might recognize a long period of time until you reach a level of good synchronization. This will happen because NTP stores the static drift of the internal oscillator locally. When this information is not available (new devices, after firmware update, using another NTP server) it must be generated at first. NTP generates this information automatically after running at least one hour being fully synchronized to a valid NTP server. From that NTP will update it each hour to improve its quality. With this information, the startup performance of NTP increases significantly resulting in high synchronization quality even within half an hour.

Nevertheless, it is recommended to run NTP as long as possible to reach the highest accuracy because it keeps track of the local clock's characteristics.

The following figure shows a typical NTP startup process of two LMGs in our testing setup synchronized to the local timeserver. At first NTP does some heavy offset corrections, and then it reaches a relatively high level of synchronization where the y-axis in this figure does not scale well anymore.

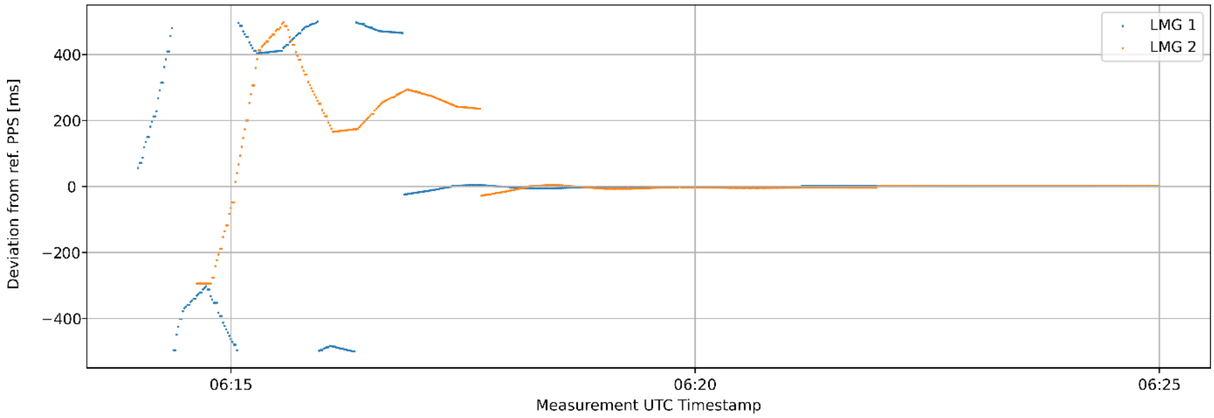

Figure 6: NTP Startup Process

What you cannot achieve with NTP

If you want to conduct NTP experiments by yourself or just check the absolute difference of the internal clock against an external time reference, there's a config value to enable the PPS output on the LMG. This is done by setting the option *CLKDISPPSDIR 1* to enable a PPS at the *reserved* pin at the sync-jack connector. PPS has its rising edge of the 5V TTL signal at the moment the internal clock of the LMG ticks up one second. Hence, it's possible to compare the signal to a reference signal and other instruments. But always keep in mind that the static delays added by different environments or devices are not predictable and could differ from the ones shown in this Whitepaper. If you want to define the exact delays in your own environment, you will have to run your own experiments and compare the PPS originating from the LMG with your own reference clocks PPS signal.

Source

Computer Network Time Synchronization: The Network Time Protocol on Earth and in Space, Second Edition. https://www.eecis.udel.edu/~mills/book.html.

ZES ZIMMER Electronic Systems GmbH
Pfeiffstraße 12
D-61440 Oberursel
Germany

Sales and Application
Tel. +49 6171 88832-0
Fax +49 6171 88832-28
E-Mail: sales@zes.com