TSEP

Technical
Software
Engineering
Plazotta

LXI Reference Design V1.30

LXI Reference Design V1.30

The LXI Reference Design V1.30 has been released and is available for all members. This version enhances the current features of the LXI Reference Design, as it adds the Extended Function “LXI Clock Synchronization” to its repertoire.

Hardware

The LXI Clock Synchronization which basically covers the IEEE1588 standard is heavily hardware depended, therefore it is crucial to have drivers available for Windows and Linux systems.

For Linux there is an open source solution available and can easily be acquired with the linuxptp package. In this case handling the ptp hardware is achieved with the commonly known functions ‘syscall’ and ‘setsockoptions’.

For Windows, however, there is no such open source solution available therefore TSEP has implemented for this purpose a 1588 stack which includes necessary drivers to access the 1588 hardware clock as well as filter and stamp incoming 1588 messages. These drivers are available for all LXI members alongside the LXI Reference Design.
The drivers support the Intel network chip families I21x and I35x.

The 1588 Algorithm

The IEEE 1588 tries to synchronise several independent clocks. Each of these clocks is in general implement as counters, which increment their counter at a given frequency. With the frequency and the counter value it is possible to derive the current time. As it is technically not possible to generate the exact same frequency by several oscillators, one has to readjust the frequency. As it is technically easier to manipulate, the counter cycle is altered. The adjustment has to be achieved by a control algorithm, as the adjustment is influenced by various disturbances. Additionally disturbances on the transport path must also be taken in account. As each IEEE 1588 implementation is based on their hardware and hardware-topology, there is no such thing as “the controlling algorithm”.

Basically any algorithm can be categorized in two groups. The first group is based on simple algorithms which concentrate on determining the error in the own clocks frequency by the determined time difference between master and slave (also known as MeanPathDelay). This type of algorithm is independent of the used hardware-topology and delivers reasonable results.
The second group are algorithms which try to determine errors in the system and include these in the calculation of the error in the own clocks frequency. These algorithms are only sensible if the to be used hardware and expected topology is

known. Due to the used hardware, fault models can be created and used. For these types of controlling algorithms Kalman-Filters are specifically suitable as they can be fitted to the particular problem.

Each controlling algorithm contains at least two states. During the first state the offset between master and slave (MeanPathDelay) is so large that the algorithm cannot close the gap in an acceptable time frame. In this state the acquired time from the master is set directly without any adjustment as the slaves' time, hoping that during the next synchronization interval the determined MeanPathDelay is considerably smaller. This state is kept until an acceptable MeanPathDelay has been achieved. This state is the default state when the clock is started or if the synchronization is lost due to problems. In the second state the actual controlling algorithm takes hold by trying to determine the correction value for the own clock and to bring the own clocks time as close as possible to the master clocks time.

The LXI Reference Design has a simple implementation added. However due to interfacing the algorithm, it can be easily exchanged with a customized algorithm.

Conclusion

The successful deployment of a 1588 implementation is not only based on using an existing IEEE1588 stack or a certain hardware. A problem-oriented approach to solving a problem is the key to success. The possibilities of IEEE 1588 is diversified. Without knowing the requirement for the accuracy for an IEEE 1588 implementation nor the hardware-topology at your disposal, it is only barely possible to provide a working system within the framework of requirements. Analysing ahead of time is crucial and requires a certain level of knowhow.

The choice of hardware is of paramount importance, as each individual component influences the accuracy differently. Especially for high accuracy in subnano areas it is essential to use fibre optical systems. The White Rabbit project performed good spadework and created necessary constraints and corresponding hardware. Also the Intel network chips can work with fibre optical PHYs. The corresponding hardware is available as consumer ware on the market.

For highly accurate systems a perfectly aligned controlling algorithm is inevitable. The choice of algorithm is not easy and requires a certain amount of knowhow. It must be adjusted to the given hardware and its topology. The implementation and simulation of the algorithm especially with highly accurate systems is not an insignificant part of the IEEE 1588 implementation. But it is also the key to success of such an implementation. For the implementation of such efficient controlling algorithms Kalman-Filters are very good. The description of the state space and the covariance of the system- and measuring noise require significant effort and time.