

Using IEEE 1588 for synchronization of network-connected devices

embedded.com/using-ieee-1588-for-synchronization-of-network-connected-devices

Embedded Staff

March 29, 2007

12 min read

When the clocks in our lives get out of sync, bad things happen—we arrive late for meetings, miss the train, or record all but the last few minutes of our favorite television show. Wouldn't it be helpful if clocks were always set to the correct time?

Actually, if they were networked together and using the IEEE 1588 Precision Time Protocol, they could all be set to same time, typically within 100 nanoseconds of each other! While that's extreme accuracy for a kitchen clock, it's perfect for clocks pumping voice, video and data through the heart of every network-connected device. And this simple protocol costs less network bandwidth and fewer dollars than other alternatives.

Specifically, the advantages of incorporating [IEEE 1588-2002 Precision Clock Synchronization Protocol for Networked Measurement and Control Systems](#) (PTP) into your network connected design include:

- * network clock synchronization accuracy in the sub-microsecond range
- * synchronization of clocks with differing precision, resolution and stability
- * low-cost implementation in multicast messaging networks such as Ethernet
- * fast re-synchronization when system changes occur
- * simple installation and maintenance.

This protocol is already widely adopted by test and measurement, power-line management and industrial automation applications. IEEE 1588 principles are adopted to synchronize clocks in networks running [Industrial Ethernet](#) protocols such as [EtherNet/IP](#) and [Ethernet Powerlink](#).

IEEE 1588 Version 2 (V2) is undergoing revision, and is expected to be published by mid-2007. It features improved support for large redundant networks and high-performance telecommunications applications.

For example, electric power utilities must synchronize across large-scale distributed [power grid switches](#) to enable smooth power transfer and maintain power supply integrity across a patchwork of systems built from many different technologies exposed to extreme heat and

cold. Reliability, durability and longevity are particularly important for power management, and timing accuracy is critical to maintain protection of the power grid. It's expensive to have a Global Positioning System (GPS) device in every substation, especially when IEEE 1588 could do the same job for a fraction of the cost.

The arrival of [femtocell home basestations](#) (Figure 1, below) will let me use my mobile phone inside my home instead of standing outdoors in -40 degrees (and I'll leave you to wonder why I don't relocate to Freescale headquarters in balmy Austin, Texas).

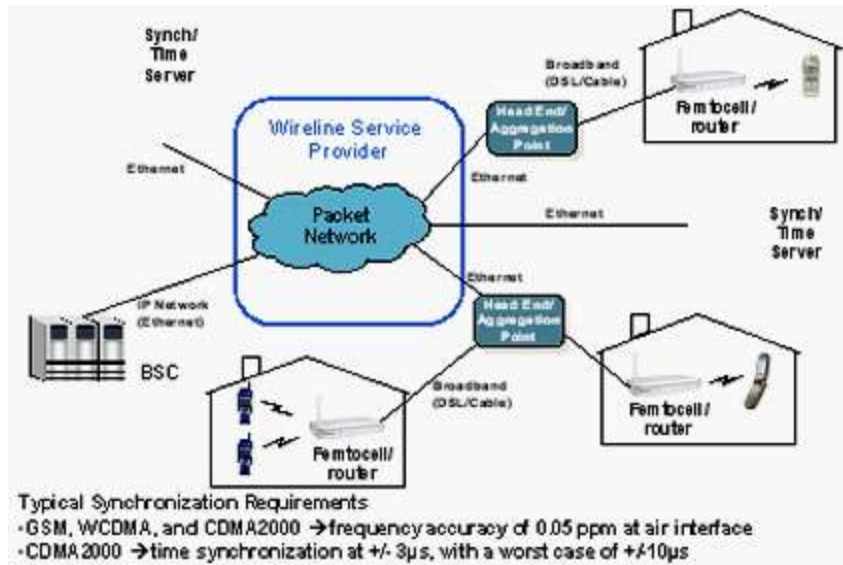


Figure1. Use of IEEE-1588 in a femtocell base station

A key ingredient for this application is a frequency-aligned, and in some cases, a phase-aligned clock that provides timing to every homebase station. Some customers consider using Network Time Protocol (NTP), GPS and other schemes to accomplish this, but many are now considering IEEE 1588 as it becomes widely supported by embedded processors targeting this high-volume application. This article covers the interesting debate in more detail below.

Interest is also growing for [telecommunications](#) applications where robust and stable clock distribution to all network nodes is required to enable the transition from leased [T1/E1 lines](#) to less expensive Internet Protocol (IP) networks.

However, many telecommunications applications are waiting for IEEE1588 V2 enhancements such as unicast messaging and shorter frame sizes to reduce the bandwidth consumed by the protocol messaging. Version 2 also introduces new PTP devices called transparent clocks that are used to prevent error accumulation in cascaded topologies, and Version 2 provides extensions to enable redundant systems.

Timing is everything The IEEE 1588 protocol enables rapid convergence (typically less than a minute, depending on the network topology) to sub-microsecond time synchronization between heterogeneous distributed devices controlled by clocks of differing resolution and stability.

IEEE 1588 can be implemented solely in software to give accuracy in the sub-100 microsecond range. This is similar to that seen with other software implemented protocols such as [Network Time Protocol \(NTP – RFC 1305\)](#) and [Simple Network Time Protocol \(SNTP\)](#) methods that operate across the same network topology.

However, if timestamping is performed in the application layer, interrupts and other unpredictable software processes can introduce jitter and latency which may impair the synchronization. Even the use of a very precise external oscillator won't overcome the stack jitter associated with a software-only 1588 implementation.

Most applications require the higher accuracy achieved by timestamping packets at the interface between the physical (PHY) and data link (MAC) layers (often referred to as “hardware timestamping”). IEEE 1588 hardware timestamping typically improves accuracy to 100 nanoseconds or better for certain network configurations, which is better than NTP, SNTP, Time-Triggered Protocol (TTP – www.ttpforum.org) and SERCOS (IEC 61491) methods.

There are several ways to implement IEEE 1588 hardware timestamping. In previous years, the only option was to connect a field programmable gate array (FPGA) between the Ethernet PHY and MAC to timestamp the arrival and departure of every incoming and outgoing IEEE 1588 SYNC and DELAY_REQUEST event message.

However, now that Freescale and other silicon vendors have implemented the hardware timestamp function on integrated communications processors, customers are adopting these new devices to significantly improve cost, power and board footprint.

As with all time synchronization methods, better clocks give better results. IEEE 1588 provides the flexibility to support different options to match most cost and accuracy requirements.

For example, the lowest cost option might use an on-chip system clock for timestamping packets. An external clock oscillator can give higher accuracy, with highest accuracy (and cost) coming from an external ovenized [voltage controlled oscillator](#) (OVXO). Protocol software supporting these three options should be easily configurable to select the desired clock source type.

Remember, no matter which implementation or clock option you select, IEEE 1588 synchronization resolution time is typically less than a minute (typically faster than NTP and

SNTP), and system cost, hardware footprint and CPU load are much lower than for all the other timesynchronization methods mentioned above.

Benefits of IEEE 1588 Version 2. Appendix A of the IEEE 1588 Version 1 (V1) specification indicates that the best PTP performance can be achieved by minimizing the number of nodes between the clock source and the slave devices.

When devices are stacked in a hierarchy, error accumulates at each level and low-accuracy intermediate devices will dominate the error experienced by the end devices.

Since hierarchical architectures are common in public and private IP networks, V2 is being developed with considerable input from experts in telecom, networking, test/ measurement, and industrial controls, who understand the special requirements of large distributed networks with many hops and inherent redundancy. This experienced team is now helping to specify support for new features such as:

- * Subnanosecond accuracy , which is especially useful for synchronizing test and measurement equipment.
- * Faster synchronization (SYNC) message rates. Unlike V1, which specifies that SYNC messages occur no faster than once a second, V2 defines a much wider range of mean SYNC message rates, even allowing for rates that are much greater than 1000 messages per second. However, most systems are expected to use mean SYNC message rates much less than this to reduce network traffic and optimize between network loading and synchronization performance. Faster SYNC message rates generally require more hardware timestamp assistance, and the CPU needs more performance to process the increased number of messages. Faster SYNC message rates are critical for telecommunications, residential Ethernet, and many control applications.
- * Shorter PTP messages, unicast messaging, new messages (path delay request/response/response follow-up) and message fields . These all reduce network bandwidth overhead and the resulting potential network queuing delays. These message optimizations are critical for telecommunications, residential Ethernet, and many control applications.
- * Transparent Clocks (TC) , which can be used to prevent error accumulation in cascaded topologies. Network-induced jitter can occur when each packet takes a slightly different time to pass through a network switch.

Resulting time sync degradation can be avoided if switches implement a transparent clock at each network hop. End-to-end & peer-to-peer Transparent Clock devices calculate the delay

incurred by a packet passing through the device (called the residence time) and update a time correction field in the PTP payload to account for this delay. Thus, each switch appears to be a “wire” which does not skew the time calculation for packets passing through them.

- * **Fault Tolerance**, which is needed to ensure no single network element failure can cause sub-tending clocks to fail. In a system with redundant grandmasters, the redundant grandmaster must be able to detect failure of the first grandmaster, and the sub-tending clocks must be able to switch from one grandmaster to another without incurring unacceptable frequency or phase jumps, or losing the clock for sub-tending nodes. Fault tolerance is critical in telecom applications and many control applications, where safety and reliability are required.

- * **TLV extension** to extend the protocol features/options. This allows the protocol to support extensions to meet the requirements of future applications.

- * **Profiles**. These define what PTP features and settings are needed for different market applications.

- * **Mapping IEEE 1588 to other transport mechanisms** such as 802.3/Ethernet, PROFINET, and DeviceNet.

More about Boundary and Transparent Clocks
Boundary Clocks were defined in V1 to support IEEE 1588 synchronization within networks containing several subnets. Boundary clocks typically have more than two ports, with one port serving as a PTP slave to an upstream clock master, and the other ports serving as PTP clock masters to downstream PTP slaves. Boundary Clocks employ a servo loop to recover the clock on the slave port, and then the recovered clock feeds Boundary Clock ports serving as PTP clock masters.

Due to the cumulative effect of multiple servo loops, cascading multiple Boundary Clocks within a network can significantly degrade clock accuracy of the system. Quality of the crystal oscillator used in a Boundary Clock can also impact the accuracy of the PTP clocks in the system. If a poor quality crystal is used in a Boundary Clock, PTP slaves attached to the Boundary Clock device will likely have inferior synchronization to the PTP clock master.

Transparent Clocks were added in V2 to update and append a timing correction field within certain messages. The correction field is updated with a value representing the “residence time” required for that message to pass through the Transparent Clock.

Here is some additional information about Boundary and Transparent clocks:

- * Boundary Clocks do not propagate Sync, Follow_Up, Delay_Req, or Delay_Resp messages.

- * Boundary Clocks are suitable for topologies with a small number of switches.
- * Cascading Boundary Clocks introduce the cascade effect. A Boundary Clock distributes timing based on its local clock, and each clock depends on the quality of all preceding clocks.
- * Boundary Clocks can cause accuracy and stability problems in highly cascaded or daisy-chained network topologies because Boundary Clocks contain IEEE 1588 control loops. Cascading Boundary Clocks is just like cascading phase locked loops in the sense that the jitter starts to accumulate down the chain.
- * Transparent Clocks do not generate timestamps. Instead they update correction fields in event messages according to the residence time of the event message within the Transparent Clock.

The correction field will accumulate the value of the residence multiplied by the number of cascaded Transparent Clocks. Ordinary and boundary clocks account for the value of the correction field in their calculations. This results in reduced jitter and increased clock accuracy and stability.

- * There are two different types of Transparent Clocks: End-to-End (E2E) and Peer-to-Peer (P2P). P2P Transparent Clocks are different than E2E Transparent Clocks in that each port on a P2P node computes the peer link path delay with its link partner on another P2P node.

The link path delay calculation is accomplished through new V2 messages: PDELAY_REQ, PDELAY_RESP, and PDELAY_RESP_FOLLOWUP (optional). P2P Transparent Clocks add the ingress link path delay time to the value in the correction field (in addition to the residence time of the message).

- * P2P Transparent Clocks only work with nodes supporting the peer delay function. This means that they work with ordinary and boundary clocks that support the peer delay function, as well as other P2P Transparent Clocks.

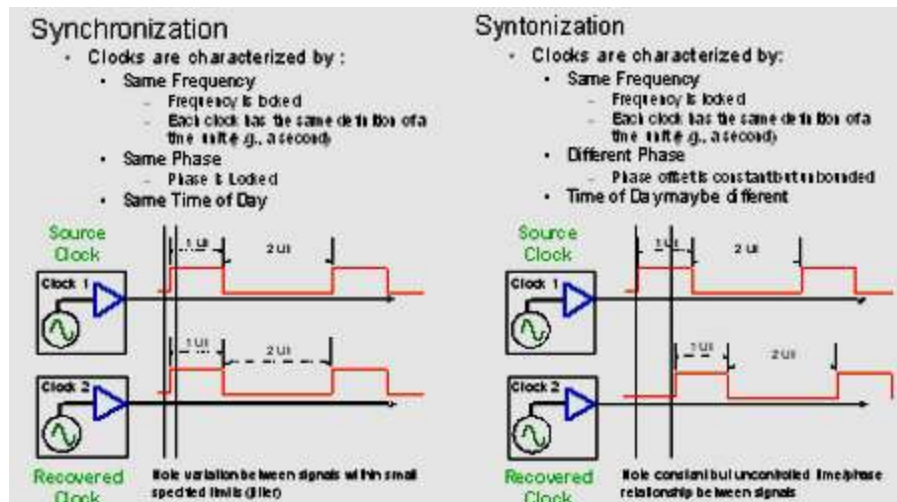


Figure2. Use of 1588 Syntonization in GSM wireless networks

Syntonization versus Synchronization

Certain telecommunications applications, like GSM wireless networks, require syntonization, where two clock frequencies are locked together but the phases may be different, as shown in Figure 2, above. This is analogous to the timing information passed along T1/E1 communications circuits today.

The clock is passed from point A to point B across the T1/E1 link, but there is no requirement to account for the propagation delay which results from the clock traveling between the two points. In other words, the difference in phase of the two clocks measured at A and B is caused by the propagation delay of the clock traveling down the link.

Other applications, like CDMA wireless networks, require clocks to be aligned in both frequency and phase, e.g., when two clocks share the same time of day and advance at the same rate. If it is important to minimize the phase difference between two points, then the delay due to the network path delay must be known.

This is typically done by passing time-of-day information between two nodes and allowing them to calculate the phase offset due to the network path delay. CDMA systems typically need a precise time of day at each node (meaning BTS) to correctly code/decode the multiplexed data. These systems often use expensive GPS receivers to get this information.

Some telecom services vendors are skeptical about relying on the IP network connecting the base stations and the Base Station Controller (BSC) to consistently deliver accurate time-of-day information to allow the data to be properly coded and decoded.

One primary concern is that customers may not control every part of the Ethernet network connecting their end-point equipment together. Public Ethernet networks are often shared by many customers and can be subject to complications that don't occur with leased T1/E1 lines.

These issues include sudden network reconfiguration, queuing delays due to traffic bursts, and delays due to traffic patterns such as heavier traffic during the day than at night.

Additionally, telecom vendors are sensitive to minimizing both equipment cost and network downtime as the transition is made from distributing timing information across T1/E1 networks to distributing timing information across an Ethernet network.

IEEE 1588 V2 seems to be a good fit for 3GPP femtocell applications, where only edge clock synchronization is required. However, it is not yet understood if it will work for all CDMA2000 femtocell applications, where clock phase and frequency synchronization is required.

Symmetrix has posted its views in an [online article](#), and Ericsson mentions a similar strategy in a document they have [posted online](#). An interesting thing to note in Ericsson's paper is that the concept of timestamping at the interface of the receiving/transmitting node is equally applicable to NTP-based networks (note the diagram in their Figure 4 where the T1 through T4 timestamp points are the classic interface points).

We might also see SNTP-hybrid networks use hardware (MAC-layer) timestamps. Even though SNTP is a very mature protocol typically used for synchronizing very remote WAN networks, nothing prohibits it from being used in a MAN/LAN environment. Interestingly, this SNTP-hybrid approach can often be supported by the same hardware timestamp mechanism added to devices to support IEEE 1588.

What IEEE 1588 solutions are available now?

Support is popping up everywhere for this protocol. The [IEEE 1588 website](#) mentions active device and software suppliers. The best IEEE 1588 hardware solutions address market requirements by offering silicon with:

- * Multiple MAC interface options, such as support for MII/GMII/RMII/RGMII
- * Flexible clocking options to support both low cost and high accuracy designs
- * Phase and frequency compensation circuitry to enable slave clock recovery
- * RX and TX hardware timestamping close to the physical interface
- * Detection of PTP frames, with a mechanism for reporting the reception of PTP frames to the CPU
- * Output trigger ports to generate PTP-time based external events
- * Input trigger ports to timestamp input events

- * PTP-time based timers to run timer interrupt functions based on synchronized timer rather frequency of the on-board oscillator
- * A one pulse per second signal output port for testing purposes

The next gathering of the IEEE 1588 development community occurs October 1-3, 2007, in Vienna, Austria. Protocol users are scheduled to report on technical issues, product development and application experience. The conference is scheduled to include a tutorial for newcomers, a plug-fest to explore interoperability of various IEEE 1588 implementations, and detailed discussion on V2. We hope to meet you there!

Editor's note: Freescale's QUICC Engine and Enhanced Triple Speed Ethernet Controller (eTSEC) technologies incorporate communications interfaces to optimize IEEE 1588 Precision Time Protocol (PTP) by time-stamping Ethernet packets at the physical/datalink layer. Both are included in PowerQUICC communications processors such as MPC8360 and MPC8313. Freescale is also collaborating with IXXAT Automation GmbH, to offer pre-configured commercial off-the-shelf total system solutions running the IEEE 1588 protocol, including the IXXAT IEEE 1588 application demonstrated on the Freescale MPC8349E PowerQUICC II Pro communications processors mid-2006.

A demonstration of the IEEE 1588 spec with better than 50 nanosecond accuracy on MPC8360 PowerQUICC 2 Pro will be performed in Avnet's Booth #624 at Embedded Systems Conference, San Jose, April 3-4, 2007. Free IXXAT evaluation software for Freescale development boards may be downloaded from www.ixxat.com. References and Additional Resources:

- 1) [Aligning system clocks over networks with the IEEE 1588 Remote Timing Standard](#) (PDF), by Alexandra Dopplinger and Jim Innis, Freescale Semiconductor and Werner Abt, IXXAT Automation.
- 2) [IEEE 1588 website](#)
- 3) John Eidson, "Tutorial on IEEE 1588," Agilent Technologies, October, 2005.
- 4) "IEEE 1588 PTP Introduction," [IXXAT Product Brief](#).
- 5) "PowerQUICC Integrates IEEE 1588 Time Synchronization," [Freescale FACT sheet](#) (PDF), July 21, 2006.
- 6) John C. Eidson, [Measurement, Control and Communication Using IEEE 1588](#), (Springer-Verlag London Limited 2006)

Alexandra Dopplinger, P.Eng., is an Industrial Segment Marketer at [Freescale Semiconductor, Inc.](#) in Ottawa, Canada. She has marketed Power Architecture technology and network processors for five years. Prior to that, she designed highly available telecom systems and holds one patent for a redundant LAN implementation.

JimInnis is a Systems Engineerfor the Digital Systems Division atFreescale Semiconductor, Inc. in Austin, TX. He has been involved inthe specification and design of PowerQUICC devices for five years.