The DOCSIS Timing Protocol (DTP)
Generating Precision Timing Services from a DOCSIS System

John T. Chapman, CTO ATTG & Fellow, jchapman@cisco.com
Rakesh Chopra, Principal Engineer, rakchopr@cisco.com
Laurent Montini, Technical Leader, lmontini@cisco.com

Abstract

New market opportunities for DOCSIS include applications such as cellular backhaul of femtocell, picocell, microcell, and macrocells. These applications may require network timing in terms of time and frequency. With the deployment of IP and Ethernet based networks, PTP (IEEE 1588) and Synchronous Ethernet have become popular approaches for distributing carrier-class network timing over a network.

DOCSIS and the HFC plant present many challenges why it is difficult to propagate network timing information from the headend, through a DOCSIS network and into a CPE device with any degree of accuracy. These challenges include:

- *HFC plant asymmetry,*
- *DOCSIS asymmetry due to the upstream scheduler variability,*
- *unknown asymmetrical plant delay between CMTS and CM,*
- *unknown delay of CMTS and CM PHYs,*
- *uncalibrated ranging.*

This paper proposes a solution called DOCSIS Timing Protocol (DTP) and discusses how DTP can address these challenges in a DOCSIS system and what specification and product changes are needed to the DOCSIS CM, CMTS, and DTI Server. The resulting design can support the generation of precision timing protocols such as NTP, PTPv2 (IEEE Std1588-2008) and Synchronous Ethernet that can serve new and evolving CPE devices with traceable time and frequency synchronization requirements.

TABLE OF CONTENTS

INTRODUCTION

Introducing DTP

The DOCSIS Timing Protocol (DTP) is the proposed name for a series of hardware, software, and protocol modifications to the DOCSIS system to support the highly accurate and traceable generation of precision timing from the CM.

The DOCSIS system in the context of DTP includes the CM, the CMTS, and the DTI Server.

Precision timing in the context of DTP refers to protocols such as Network Time Protocol (NTP), Precision Time Protocol (PTP), or hardware interfaces such as a Pulse Per Second (1PPS) output and Synchronous Ethernet.

Any specific reference to supporting a PTP output on the CM in this white paper inherently could include other protocols such as NTP and/or 1PPS-style interfaces and their associated protocols. A standard telecom 1PPS output is not defined at this time but is under study by the ITU-T.

All references to PTP and IEEE 1588 imply the latest version of PTP that is PTPv2 as defined by IEEE Std 1588-2008.

Wireless Basics

For clarification, it is useful to review the basics of wireless terminology for this paper.

Wireless includes cellular technology such as LTE, GSM/UMTS and others as well as non-cellular technologies such as Wi-Fi and DECT. This white paper targets cellular wireless technologies since they generally require highly precise time and frequency synchronization.

Each wireless technology generally has a base station that acts as a coupling point between the wireless and wired network. The classification of base station is related to its coverage and usage. Common classifications used today along with typical coverage and typical usage areas are listed on the next page. Each technology and cell site produces slightly different results.

Standard GSM macrocell range is limited to 35 km but an extended range cell may go up to 60-100 km in certain areas. Range depends on various parameters including technology, power, area or coexistence with other cells of same or distinct radio technology. The example ranges below are for WCDMA in open air (reduced coverage inside of buildings)

- Macrocell
  - WCDMA: 43 dBm/30m = 1 km
  - Rural areas or along highways
- Microcell
  - WCDMA: 33 dBm/20m = 400 m
  - Malls, hotels
- Picocell
  - WCDMA: 24 dBm/10m = 200 m
  - Transportation hub, airplane
- Femtocell
  - WCDMA: 24 dBm/1m = 71m
  - Actual coverage area is usually less due to being inside of a building.
  - Residential home

Mobile Backhaul Synchronization

In the last few years, synchronization in access networks has become an important topic because of the evolution from TDM-based to packet-based networks. In particular, the mobile wireless operators are struggling to increase their backhaul capacity that is required by the newest radio technologies in order to provide greater bandwidth and improved services.

Although the introduction of smaller capacity base stations (namely microcell, picocell and femtocell) permits mobile and broadcast operators to improve the wireless service by providing better coverage, it also demands increasing the number of network connections.

Such trends lead to optimization of the mobile backhaul infrastructure.

Packet-based networks allow the operators a cost-effective way to fulfill the necessary improvements in bandwidth, coverage and access. Today, IP over Ethernet is the most utilized transmission option for the aggregation networks.

The choices for the last mile access transport technologies include Ethernet (fixed and microwave), Passive Optical Network (PON), Digital Subscriber Line (DSL) or legacy TDM line such as T1/E1 or SONET/SDH (either fixed or microwave). Cable HFC (hybrid fiber coax) networks are being considered also as the bandwidth of DOCSIS (Data over Cable Service Interface Specification) based systems continue to increase.

The first generation of this mobile backhaul network evolution called for the support of legacy TDM circuits (e.g. for 2G GSM base stations). Replacing these circuits created a market for circuit emulation service over packet networks. Because T1/E1 requires accurate and stable clocking, circuit emulation services (or TDM pseudo-wires services) over packet networks inherited the need for frequency synchronization.

DOCSIS 1.1 introduced circuit emulation support and was able to leverage inherent DOCSIS frequency transfer such as NCR (Network Clock Recovery) via Symbol Clock Lock. Other access technologies such as SHDSL or GPON also have bit timing (physical layer) capability. But this requires equipment at both ends to be able to either receive or to retransmit clock signal. Timing distribution must then be accordingly planned.

Classic Ethernet has no such synchronous clocking capability. Adaptive

Clock Recovery (ACR) – that is, recovering frequency from a packet flow – from Circuit Emulation Services (CES) traffic was the first method developed to support TDM pseudowires [G.8261]. If such a solution was sufficient in some cases for CES application, it appeared to be sub-optimal to support base station radio interface requirement.

Radio Frequency Synchronization

Indeed, one critical aspect of mobile base stations (from GSM to LTE or WiMAX) and broadcast transmitters is their need for synchronization of their radio interface. Accurate frequency synchronization between base stations allows user handsets to seamlessly handover between base stations, reduces interference between cells and optimizes radio bandwidth capacity.

To improve timing services available from networks, particularly Ethernet based, ITU-T Question 13 in Study Group 15 took the leadership on investigating solutions and defining the appropriate specifications. Focus was first given to frequency distribution because of the CES application and 2G/3G base stations.

This focus led to adopting Synchronous

Radio Technology or Service	Cell (Base Station) Type	Frequency Read: better than…	Phase or Time Synchronization Read: less than…
GSM	Macro Pico	±50 ppb ±100 ppb	N/A
WCDMA (and LTE) FDD	WideArea Medium/LocalArea (micro/pico-cell) Home BS (femtocell)	±50 ppb ±16 ppb (OBSAI) ±100 ppb ±250 ppb	N/A
WCDMA TDD	WideArea LocalArea	±50 ppb ±100 ppb	±2.5 µs between base stations
TD-SCDMA	WideArea LocalArea	±50 ppb ±100 ppb	±3 µs between base stations
LTE TDD	WideArea LocalArea	±50 ppb ±100 ppb	±3 µs between base stations (may range from ±0.5 µs to ±50 µs)
CDMA2K	Macro Pico and Femto	±50 ppb ±100 ppb	ToD (UTC) sync *should* be less than 3 µs and *shall* be less than 10 µs
WiMAX Mobile		Up to ±1 ppb (with an average target of ±15 ppb)	Usual values between ±0.5µs and ±5 µs
MB SFN Service		±50 ppb	±1 µs
LTE-Advanced Services (CoMP, relaying function, carrier aggregation…)		Up to ±5 ppb (CoMP)	±0.5 µs [±1 µs] may be < ±0.2 µs (TBC)
DVB SFN		Up to ±1 ppb	General agreement: ±1 µs

Table 1 – Cellular Accuracy Requirements

Ethernet, a physical layer method that demands hardware changes in Ethernet equipment as well as to define recommendations to support packet-based frequency transfer with no hardware changes in packet network elements.

As mentioned earlier, frequency synchronization (also named syntonization) is a common requirement for base stations. Refer to Appendix IV of [G.8261]

Table 1 presents a summary of the main wireless applications driving the standard development for synchronization in telecom operations. These telecom network requirements are currently the tightest known and therefore provide guidance for what accuracy levels DTP should target.

Because of smaller impact of the Doppler effect, smaller cells sites can tolerate more relaxed requirements as shown in Table 1.

Phase/ToD Synchronization

Table 1 also presents another critical aspect of some mobile base stations: the need for phase or time of day (ToD) synchronization. Most of the values in Table 1 are based upon publicly available information and standard references. Some values depend on radio parameters and a few, such as CoMP, have to be confirmed by the appropriate organization. ITU-T WG15 Q13 is currently working on [G.8271] that will further describe the requirements and point to appropriate references. Similar phase or ToD requirements are also seen in other market segments such as: broadcast operators, power utilities or Smart Grid, real-time applications as for audio video bridging, or more conventionally for better network performance measurement.

As for frequency synchronization, those wireless phase/ToD requirements apply to the radio interfaces, particularly if Time Division Duplexing (TDD) is being used (e.g., WCDMA or LTE TDD). TDD is a method allowing radio interface to transmit and receive in different time slots on the same media or frequency band.

The phase/ToD synchronization requirements for radio services can be independent of the radio transmission methods. For instance, Single Frequency Network (SFN) is used for Multicast and Broadcast Services and MBS (also named Multicast and Broadcast Multimedia Services –MBMS– in 3GPP). Phase synchronization allows the simultaneous transmission of the same frame by multiple base stations or transmitters in the same SFN domain.

In most cases, the synchronization accuracy expected from the network for large cells will typically be in the sub-microsecond range.

Other applications are less demanding. We could then categorize all these applications at different levels based upon their timing performance requirements. The lowest level may not require the same network changes as the higher levels. Table 2 proposes such a performance scale.

The level of time synchronization that can be achieved over DOCSIS without DTP depends on multiple variables such as the location of timing source, the HFC plant configuration, the protocol and its setup or the receiving clocking servo. For instance, a software-based standard NTP implementation is not expected to provide sub-millisecond time accuracy.

Level of Accuracy	Typical Applications	Range of Requirements
1	Billing, Alarms	> 1 ms
2	IP Delay monitoring (range depends on network and applications)	few µs to hundreds of µs
3	Radio interfaces requirement (range depends on technology and radio configuration) Power Utilities, SmartGrid, Real–Time Audio and Video (Broadcast, AVB)	1 µs to few µs
4	Wireless services (e.g., CoMP, LBS or E911)	< 1 µs

Table 2 - Ranking Different Applications

What about GPS?

Before network-based precise timing distribution in a telecom network became a critical development topic, the only solution was to utilize over-the-air PNT (Positioning Navigation and Timing) solutions, particularly the well-known GPS (Global Positioning System). GPS receivers were embedded in the base stations (e.g., CDMA™2000) or installed at the cell site to feed the base station (e.g., WiMAX TDD or DVB).

GPS (or equivalent GNSS – Global Navigation Satellite Systems), despite being ubiquitous, have some drawbacks. For carrier-class timing purposes, such a system can be expensive because of the installation and operation costs. Indeed, the increased number of cells would just augment the number of GPS receiver installation or call for further cabling requirements in order to distribute the GPS clock signal within a building (as for picocells and femtocells).

Moreover, multiple governmental, industry or engineering organizations have pointed out the usual over-reliance on GPS for critical public services, while highlighting that. GPS signals are susceptible to threats such as jamming (intentional or not) and spoofing. Hence, when relying on a GNSS solution, proper backup mechanisms are desirable. Currently, most of the time, an expensive oscillator or an atomic clock provides the necessary stable local reference required for carrier-class timing.

For these reasons, GPS and other GNSS, cannot be considered the only alternatives anymore. Therefore stronger attention has been given to network-based timing solutions for backing up or replacing GNSS receivers.

Alternative to GNSS solutions

For instance, one alternative way to backup a GPS receiver is to provide a stable frequency reference such as a signal traceable from a PRC (Primary Reference Clock – ITU-T) or PRS (Primary Reference Source –Telcordia/ATIS) device. Such a signal can replace the expensive local oscillator that would take over in case of GPS signal failure. The last valid time information would be maintained with a physical layer PRC/PRS-traceable signal stability.

Another option would be to provide another time reference from the network, complementing or replacing the PRC/PRS-traceable frequency source. Eventually, this network timing reference may become the principal and unique time reference available to the applications.

In summary, multiple applications with different requirements may benefit from timing services from packet networks. For the most demanding applications, such as mobile wireless, the network must provide specific support leading to improvements in transmission technologies. For naturally asymmetrical access networks, specific techniques must be developed and adopted as part of timing network engineering.

The next section will present new technologies developed mainly for Ethernet networks that can be used as part of the cable operator aggregation network.

SYNCHRONOUS ETHERNET

TDM networks were designed and optimized to carry continuous rate traffic. Over time, they have been adapted to carry packetized IP traffic. Time Division Multiplexing (TDM) network deployments relied on Layer 1 frequency distribution techniques to synchronize multiple network elements together allowing for slower speed interfaces to be multiplexed together from multiple sources.

The master/slave synchronization architecture of hierarchical TDM networks allowed network providers to rely on all nodes of their network to be synchronized to a PRC/PRS. Over time they built up infrastructure and deployment models around this capability.

Ethernet, meanwhile, was designed to carry packet data across the shared medium of a local area network. Within each network element that linked isolated local area network segments, packets are buffered and retransmitted. This removes the requirements for frequency synchronization allowing each Ethernet node to run asynchronously from all other Ethernet nodes.

Owing to the shared medium nature of Ethernet, transmitters only send data when necessary. This technique frees the medium for other nodes to transmit. Because of this, the nodes are not constantly driving bits onto the wire, thus precluding the Ethernet link partner from continuously recovering the frequency of the transmitter.

Modern day Ethernet provides a non-shared medium with full-duplex options over

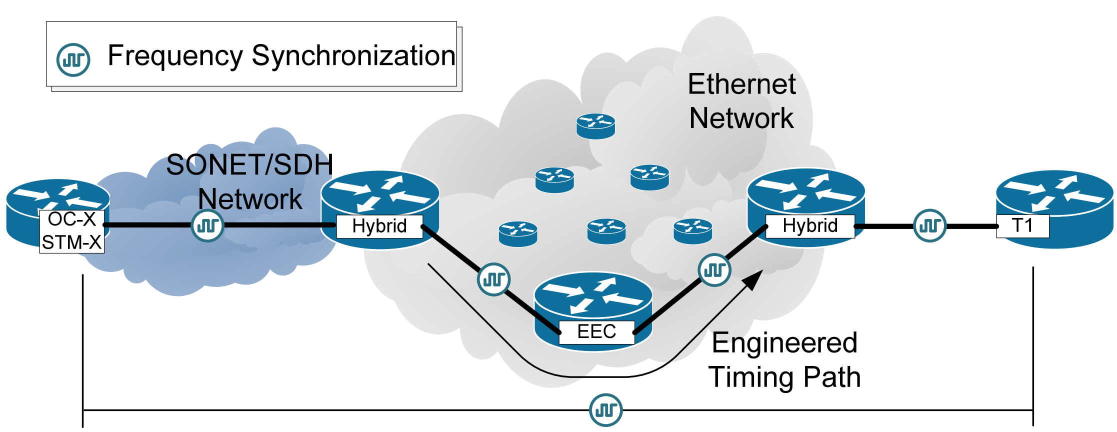


Figure 1 – Synchronous Ethernet Frequency Distribution

both fiber and copper. With these changes, the receiver can see a continuous bit stream and then can reliably recover the frequency from the transmitter allowing for Layer 1 frequency distribution.

First defined in G.8261 (2006), then complemented by ITU-T G.8261(2008), G.8262, G.8264 and a new release of G.781, Synchronous Ethernet specifies not only the method and requirements for frequency recovery and transmission but also provides standardization on advertisement of clock quality through the network.

Like all Layer 1 frequency synchronization techniques, all network elements between network segments must be capable of recovering and passing the frequency downstream. Therefore, changing a path from Ethernet to Synchronous Ethernet requires all nodes in-line to be changed to Synchronous Ethernet Equipment Clock (EEC). However, unlike traditional TDM networks where all nodes must be synchronized, only the non-Synchronous Ethernet network elements involved in the engineered timing path need to be upgraded.

Figure 1 provides a high level view of a Synchronous Ethernet network being used to frequency synchronize a SONET/SDH network to a T1 node.

With Synchronous Ethernet, network providers can replace old TDM equipment with more cost effective, higher performance, IP optimized Ethernet equipment while still enabling deployments that require frequency traceability.

For early deployment of frequency transfer, the drawback of Synchronous Ethernet is to ask for some hardware changes. Before approval of Synchronous Ethernet technology, packet-based solutions were already investigated.

IEEE 1588

IEEE 1588 standardizes the Precision Time Protocol (PTP) which is a two-way time transfer (TWTT) protocol. Another example of an earlier TWTT protocol is the IETF NTP (Network Time Protocol).

A TWTT protocol uses bi-directional traffic flow between a master or server and slave or client to exchange four timestamps, T_1, T_2, T_3 and T_4 (see Figure 2).

A slave or client clock servo will use those timestamps to synchronize as accurately as possible to the master or server clock. Refer to Appendix XII “Basic Principles of Timing over Packet Networks” of [G.8261] for further details.

While Synchronous Ethernet provides frequency traceability with Layer 1 connectivity, IEEE 1588 (like NTP) can provide time synchronization between two nodes across a packet network without mandating all intermediate nodes being replaced. Because time advances at a specific rate it is possible to also synchronize frequency with IEEE 1588 (or NTP).

IEEE 1588 also specifies system properties necessary to PTP for optimized time recovery.

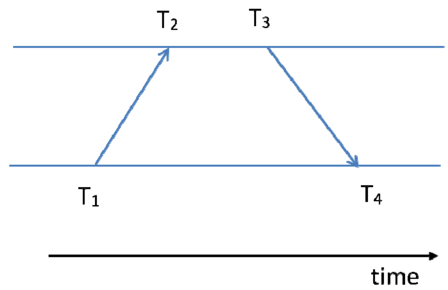


Figure 2 - Two-Way Time Transfer Principle

Evolution History

The first release of IEEE Std 1588 (PTP Version 1) was approved as a standard in 2002 and is used primarily today for industrial automation and test and measurement fixtures. The second release of IEEE Std 1588 (PTP Version 2), started in 2005 and approved in 2008, provided several key enhancements and added flexibility to the standard, enabling it to be adapted to other industries such as telecommunications.

Some of the main improvements in IEEE Standard 1588-2008 are:

- Higher packet rates for increased frequency accuracy and resiliency against packet delay variation (PDV) per G.8260
- Support for unicast transmission
- Support for redundant configurations to allow for increased fault tolerance
- Introduction of transparent clocks
- Configuration options and profiles.

PTP is quickly becoming the industry standard for highly accurate time distribution when other sources such as GPS are not available.

Network Node Types

To help define the equipment that 1588 protocol messages traverse, the standard specifies the following node types:

1. Grandmaster Clock (GM): The ultimate master of time for clock synchronization within a single PTP domain.
2. Ordinary Clock (OC): A node with a single PTP port in a domain that maintains the time used within that domain. There are two states of ordinary clocks:
   - Grandmaster Clock: A node that sources time to one or more slaves.
   - Slave Clock: A node that receives time from a master port.
3. Boundary Clock (BC): Multiple PTP ports in a single PTP domain with one slave port and at least one master port. A boundary clock can become a grandmaster clock.
4. Transparent Clock (TC): A device that modifies PTP event messages as they traverse through it. The transparent clock calculates the time the PTP event message takes to pass through the node and stores this value into the message. By doing this the node can look "transparent" and the node's contribution to PDV can be compensated for by the slave port.
5. Management Node: A device that

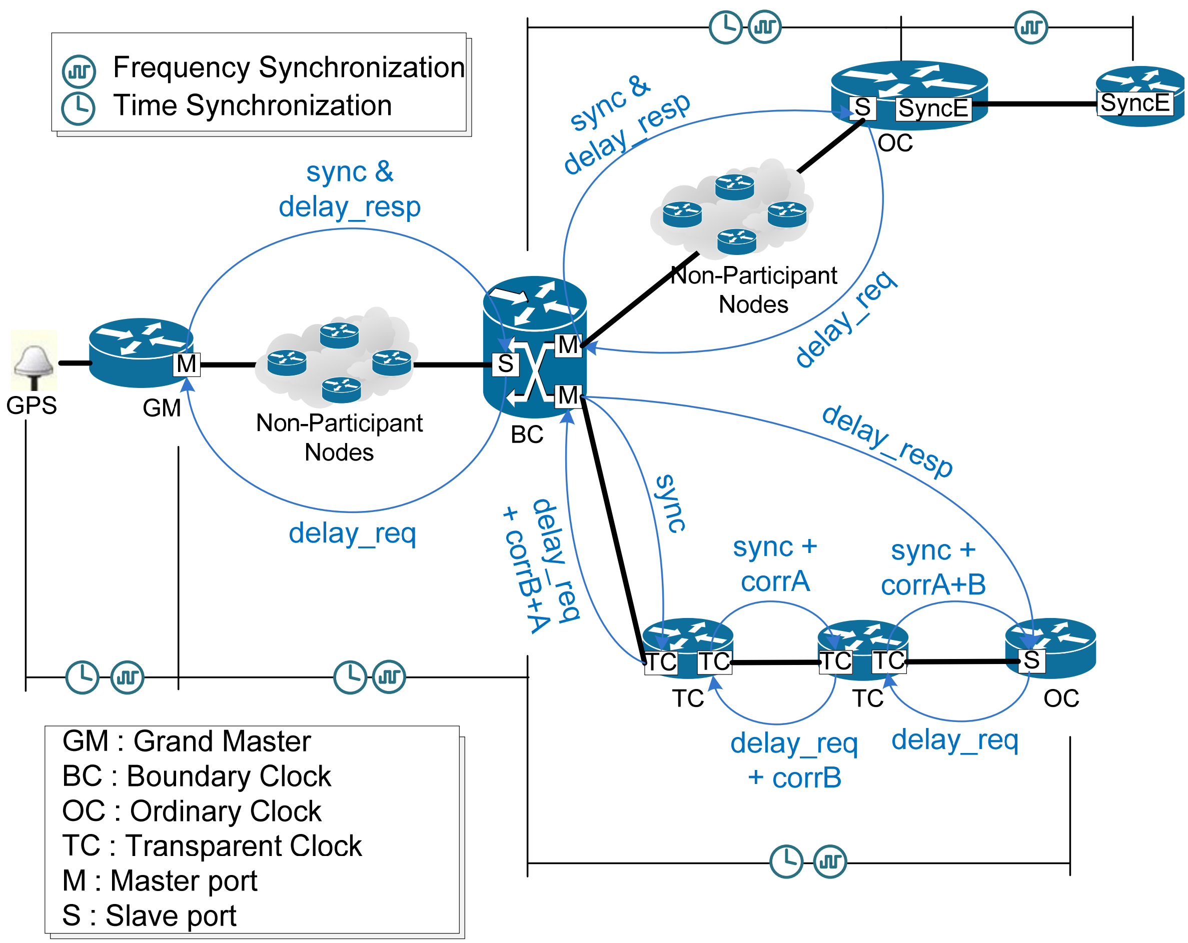


Figure 3 - 1588 Time and Frequency Synchronization

configures and monitors clocks.

Despite not being specifically defined by IEEE Std 1588, the protocol implicitly allows messages to pass through a network element that does not generate, modify, or consume PTP messages. These nodes are commonly referred to as Non-Participant nodes. These can have an impact on the recovered timing signal at a slave port due to large PDV.

Examples of these node types are illustrated in Figure 3. In the example, the grandmaster clock (GM) on the left receives its time and frequency source from a GPS receiver. It uses PTP to synchronize the slave port of the downstream boundary clock (BC) across a non-1588 aware network. The boundary clock recovers the time and frequency from PTP messages thereby roughly synchronizing it to the GM.

In addition to the slave port, the BC provides two master ports. The top master port uses PTP to allow the top OC to synchronize to its clock across another network built with non-participant nodes. The top OC recovers the time and frequency from PTP and it has a transmit reference for its Synchronous Ethernet port. The GPS and the top right Synchronous Ethernet nodes are now frequency synchronized to within a small margin of error in the short term but highly accurate and stable in the long term.

The lower master port uses PTP to synchronize the lower OC to itself. Along the path, the TC modifies the time critical events (Sync, Delay_Req) with the time the message took to pass through the TC node. The lower right OC can then recover the time and frequency from the BC and utilize the information provided by TC nodes to compensate for the PDV.

Once up and running the GM, BC, and the two OC clocks are all time and frequency synchronized to within a small margin of error. The Synchronous Ethernet node is roughly frequency synchronized to the GPS receiver.

PTP Protocol Overview

PTP defines many different message types to achieve time synchronization and node management. These are:

1. Sync, Follow_Up, Delay Request (Delay_Req) and Delay Response (Delay_Resp): These messages are utilized by PTP ports on OC and BC to synchronize time and frequency.
2. Path Delay Request (Pdelay_Req), Path Delay Response (Pdelay_Resp), Path Delay Response Follow Up (Pdelay_Resp_Follow_Up): These messages are used to measure delays between adjacent nodes when peer delay mechanism is used between master and slave ports.

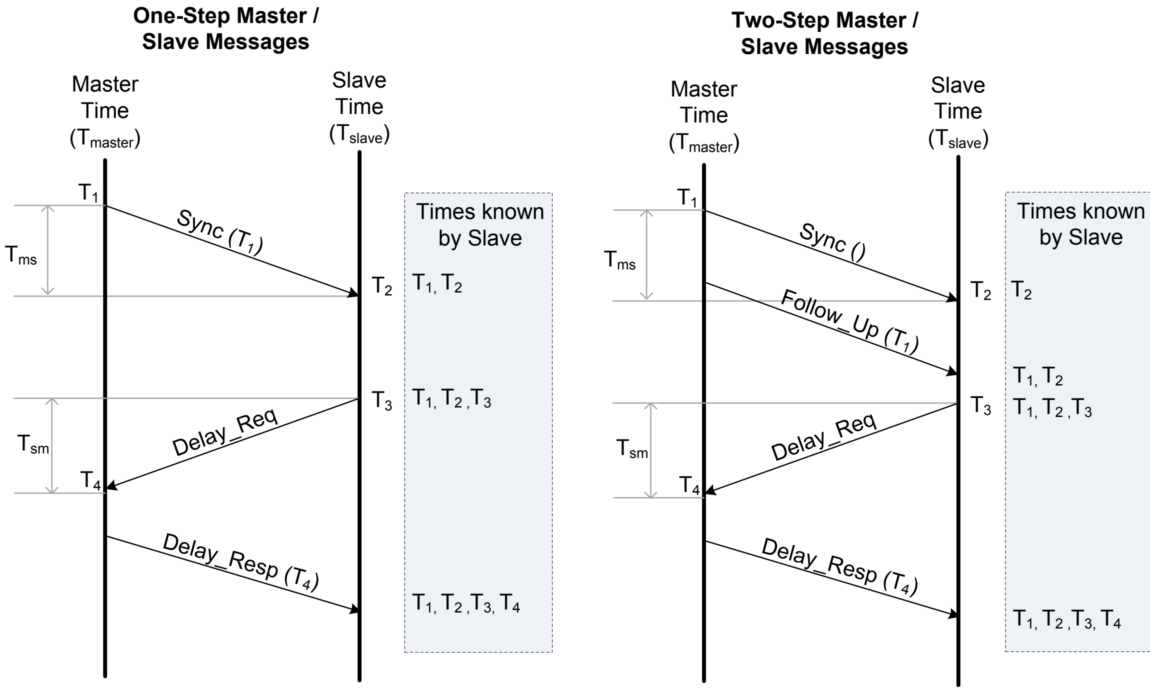


Figure 4 – PTP Message Exchange between Master and Slave

3. Announce: These messages are used as part of the clock selection algorithm.
4. Management: These messages are used to configure and monitor the PTP nodes.
5. Signaling: These messages are used for communication between PTP nodes.

We will focus primarily on the Sync, Follow_Up, Delay_Req, and Delay_Resp messages in this white paper since these messages provide the method for synchronizing time and frequency between the master and the slave nodes.

Figure 4 provides a high level logical view of the network elements and the messages used for achieving clock and frequency synchronization.

Figure 5 provides a high level view of the usage of these messages for both one-step and two-step masters.

The following provides an overview of the message usage:

1. The master port sends a Sync message containing the time T_1 when it left the master. There are two options for providing the time T_1 to the slave port.
   - Placing the timestamp T_1 into the Sync message (one-step). This requires hardware modification of the packet near the physical port to achieve a high level of synchronization.

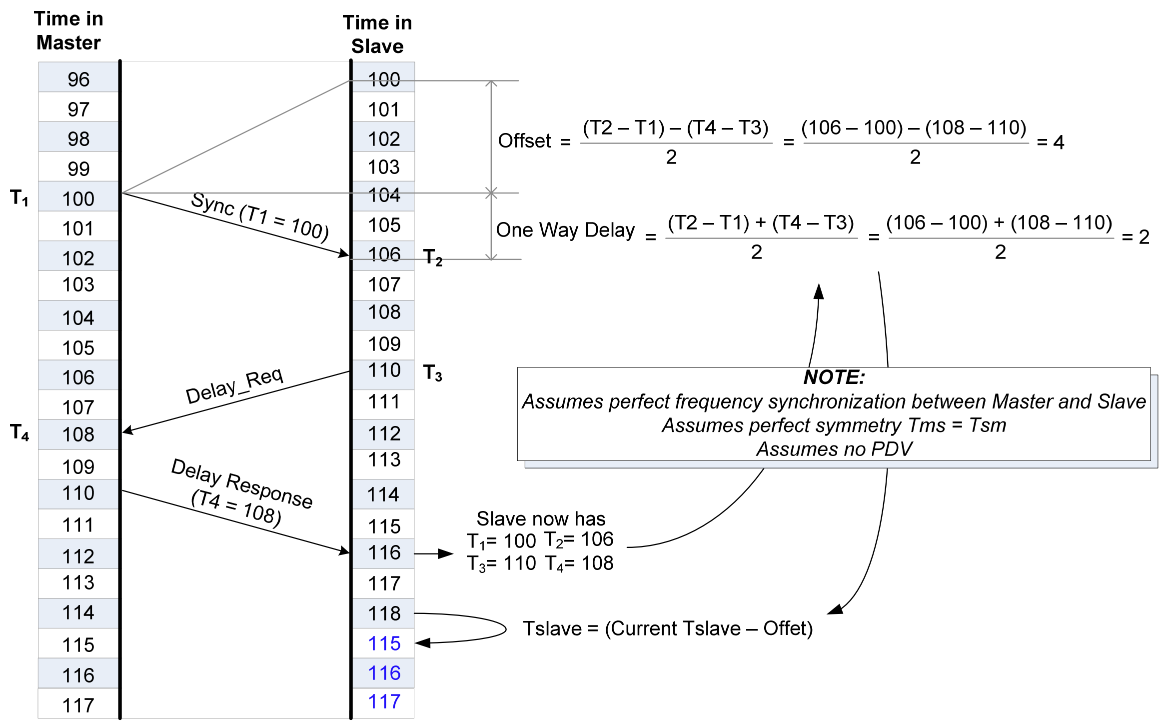


Figure 5 – Idealistic example of Slave Time Synchronization

- Placing the timestamp T_1 into a Follow_Up message (two-step). This allows the master to simply record the departure time of the Sync message. Ideally this timestamp location is near the physical port to achieve high levels of synchronization. A Follow_Up message is then generated with the Sync message departure time. This alleviates the need to do on the fly packet modification.

2. The message takes T_{ms} to travel through the network to be received at the slave.
3. The slave records the time T_2 that the Sync message arrives at its input port. If the master is a one-step master, the slave knows the time T_1 with the decoding of the Sync message. If the master is a two-step master, the slave receives the time T_1 in the Follow_Up message.
4. At this point, the slave knows the time at the master but with an unknown offset of T_{ms}.
5. To measure the delay, the slave device sources a Delay_Req message to the master. The slave records the time T_3 that the Delay_Req left the slave device.
6. The message takes T_{sm} to travel through the network to be received at the master.
7. The master records the arrival time T_4 and relays the received time back to the slave in the Delay_Resp message.

This set of message exchange provides enough information to the slave to approximately synchronize the slave to the master, as described in the following sections. However, multiple sources of error complicate the slave time recovery process.

Achieving Frequency Synchronization

Without frequency synchronization, the master and slave nodes' clocks will drift between message updates. Because these Sync messages are sent repetitively, the slave is able to calculate the drift between the master clock and slave clock with the following formula:

$$Drift = \frac{\triangle T_2 slave - \triangle T_1 slave}{\triangle T_1 slave}$$

Where ΔT_2slave is the time T_2 between multiple Sync messages being received at the slave, and ΔT_1slave is the difference in time T_1 between the same Sync messages received at the slave.

By comparing drift over time, the slave can synthesize a frequency that tracks the master frequency and keeps the time synchronized in between message updates.

Frequency recovery is only required if alternate frequency traceability methods – such as Synchronous Ethernet – do not exist and the necessary levels of time accuracy require it. If Layer 1 frequency traceability is available, it should be used since it provides a higher level of frequency accuracy and stability.

Achieving Time Synchronization

Once frequency synchronization has been achieved, time synchronization can begin. [1588 Applications] [1588 Tutorial] Before time synchronization has occurred, there is a natural offset between the master and slave devices which can be represented with two equations:

$$T_2 - (T_1 + T_{ms}) = Offset$$

$$T_4 - (T_3 + T_{sm}) = - Offset$$

Where T_{ms} is the master to slave delay and T_{sm} is the slave to master delay. Combining these two equations we get:

$$Offset = T_2 - (T_1 + T_{ms}) = - (T_4 - (T_3 + T_{sm}))$$

We now have two unknowns, T_{ms} and T_{sm}, and a single equation. To solve the equation, PTP assumes that the delays from the master to the slave and from the slave to the master are perfectly symmetrical ($T_{delay} = T_{ms} = T_{sm}$).

$$T_2 - (T_1 + T_{delay}) = - (T_4 - (T_3 + T_{delay}))$$

Solving the above equation for the one way delay (T_{delay}) we get:

$$Tdelay = \frac{(T2 - T1) + (T4 - T3)}{2}$$

By substituting in T_{delay} from above, we can solve the original offset equation:

$$Offset = T_2 - (T_1 + T_{ms})$$

$$Offset = T_2 - (T_1 + T_{delay})$$

$$Offset = T2 - \left(T1 + \frac{(T2 - T1) + (T4 - T3)}{2}\right)$$

$$Offset = \frac{(T2 - T1) - (T4 - T3)}{2}$$

The time at the slave time (T_{slave}) can then be set at some point later by adjusting the current time (Current T_{slave}) with the calculated offset:

$$T_{slave} = Current\ T_{slave} - Offset$$

Alternately, the slave time could be adjusted on the next T_2 time with:

$$T_{slave} @ T_2 = T_1 + T_{delay}$$

Time recovery can require a very complex algorithm that is affected by many real world effects such as slight variations in frequency, PDV, and network asymmetry. However in order to simplify the understanding, these real world effects are ignored in the example in Figure 5 to show how the previous math can be applied.

Time Synchronization Error Sources

There are three main sources of error for time synchronization in any TWTT protocol including PTP. These are:

Fixed Path Asymmetry

Because PTP assumes that the master to slave and slave to master paths are perfectly symmetrical, any asymmetry in the paths will result in a time offset between the master and slave nodes equal to the following basic formula:

$$Terror = \frac{Tms - Tsm}{2}$$

In the previous overly simplistic example, if $T_{ms} = 4$ and $T_{sm} = 2$ the slave would have calculated the offset as 5 instead of 4, thus leading to a time shift of 1 at the slave.

The asymmetry can arise from many sources, including but not limited to:

- network topology differences
- timestamp location differences within the master, slave, or transparent clock nodes
- node delay asymmetry through non-participant nodes.

Packet Delay Variation (PDV)

Because time synchronization relies on constant flight time between the master and slave, any variability in packet delivery in either direction will make it more difficult for the slave to accurately recover time and frequency. Each calculation of drift, offset and one-way delay will produce unique results based on the PDV in the network.

Therefore, slaves use a slave servo algorithm to integrate the results to determine the true offset and one-way delay measurements over time. Alternatively a slave servo algorithm could pre-process the time values before calculating the offset, drift, and one-way delay looking for minimum packet delays.

However as mentioned earlier this algorithm is left to the implementer and not standardized as part of IEEE 1588.

Frequency Drift Between Master and Slave

In between time updates from the slave servo algorithm, PTP time is advancing based on the slave's holdover frequency. If the frequency at the master and the slave are not perfectly synchronized the time at the slave will drift away from the master's time. The rate of drift is proportional to the frequency difference.

If the frequency on the slave is recovered from packet timing flow, as with PTP (but also true for CES or NTP), then the accuracy of the frequency recovery will be impacted by the PDV through the network.

Improving Packet-Based Timing Accuracy

IEEE 1588 transparent clocks can be used in-line between a master port and slave port to provide PDV information to the slave. This enables an increase to the maximum number of nodes between a master and a slave with the same accuracy, or to increase the accuracy of time alignment between them with the same number of nodes. Because transparent clocks record the residency time of a packet, any error in timestamp location, frequency offset or drift will have a negative impact on their performance.

IEEE 1588 boundary clocks (or NTP stratum servers) can be used to divide the PDV effects into smaller segments and to increase the scale of 1588 deployments by distributing the burden of packet generation to multiple nodes within the network. Unfortunately, because boundary clocks are susceptible to the same error contributions as a slave, they may actually have a negative impact on time alignment through the network. In between message updates, the boundary clock operates in holdover and therefore induces time and frequency error proportional to their onboard oscillator quality and slave servo algorithm.

Even with these techniques, achieving highly accurate results with IEEE 1588 requires very careful planning and implementation.

DTP OPERATION

The DOCSIS Timing Protocol (DTP) is a series of extensions to the DOCSIS protocol and the implementations of the DOCSIS CM, CMTS, and DTI Server that are intended to support protocols like PTP with a much higher degree of accuracy by leveraging the internal precision timing of the DOCSIS system.

The basic design of DTP involves synchronization of frequency and time (phase).

- Frequency is addressed by coupling the cable modem (CM) Ethernet timing to the DOCSIS downstream baud clock.
- Time is addressed by coupling the CM PTP timestamp message to the DOCSIS SYNC message timestamp.
- Time offset and asymmetry will be addressed through measurement, signaling, and ranging.

The CM would have an Ethernet output that support synchronous Ethernet [G.8261], that would have an output circuit for precision packet time stamping [802.3bf], and would support a network timing protocol such as PTP [1588].

System Description

The DTP system is shown in Figure 6. The system consists of four main components

- The CMTS. This can be an integrated CMTS (I-CMTS) or a modular CMTS (M-CMTS).
- The remote CM. This CM provides precision timing to an external entity.
- The reference CM. This is a reference CM that is identical (same manufacturer, model number, and software load) to the remote CM. It is co-located with the CMTS and is used for comparative measurements.
- The DTI Server. The DTI Server is common in M-CMTS systems. It may be external or embedded in the CMTS. In DTP, the DTI Server serves as a source of clock. It also provides measurement functions using a PTP slave port.

The additional functionality defined by DTP for the DTI Server may also be native to the CMTS.

Figure 6 also introduces various system delays that are further defined in Table 3.

CM Frequency Synchronization

DTP specifies that the CM design will synchronize the Synchronous Ethernet port to the baud clock of the downstream QAM signal. Since the jitter of the downstream DOCSIS baud clock generally exceeds the jitter requirements for Ethernet, a PLL with

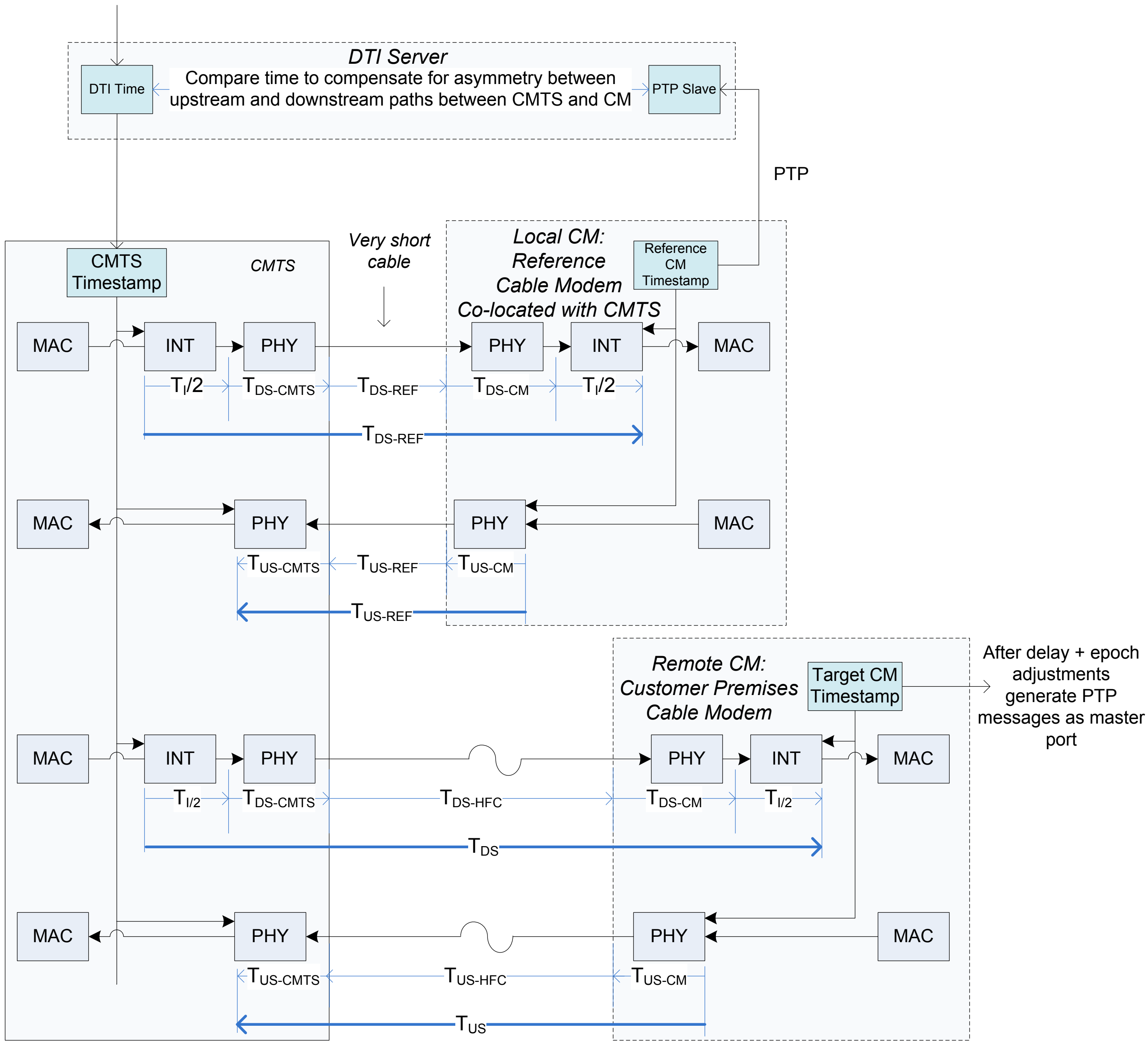


Figure 6 – DTP System Diagram

M/N frequency correction and jitter filtering will be needed.

Prior to DOCSIS 3.0, a CM in advanced time division multiple access (ATDMA) mode was not required to lock to the downstream baud clock. In TDMA or ATDMA, the CM timing was derived from entirely from the DOCSIS timestamp. In synchronous code division multiple access (SCDMA), the CM always locks to the downstream baud clock. In DOCSIS 3.0, the CMTS publishes a bit in the MAC Domain Description (MDD) message called the symbol clock-locking indicator. If this bit is set, then the CM must lock to the downstream baud rate clock.

Reference Section 6.4.28 “MAC Domain Descriptor (MDD)”, subsection 6.4.28.1.10

"Symbol Clock Locking Indicator" and section 7.1.2 "CM Synchronization" of [DOCSIS MACUP] for more detailed information.

For Synchronous Ethernet, the CMTS must set this bit and the CM must lock to the downstream baud clock for any upstream multiple access type in use, including TDMA, ATDMA, and SCDMA.

The base clock frequency of a gigabit Ethernet port is 125 MHz. The base frequency of a 100BaseT port is 25 MHz. The base frequency of the DOCSIS QAM signal is 10.24 MHz. The mathematical relationships between these clocks are:

- 10.24 MHz * 3125 / 256 = 125 MHz
- 10.24 MHz * 625 / 256 = 25 MHz

A fractional M/N PLL on the CM would implement this function.

CM Time Synchronization

Time synchronization refers to the generation of a PTP compatible timestamp at the Ethernet interface that is synchronous and phase aligned with the DTP timing source. The DTP timing source is then offset from a defined epoch. An epoch is the origin point in time of a time scale.

The DOCSIS timestamp in a stand-alone CMTS has an arbitrary value. If the CMTS is connected to a DTI Server and the DTI

Variable	Known	Comments
Tds	No	Total downstream delay from CMTS timestamp reference point to the CM timestamp reference point. This is the ultimate value that needs to be determined.
Tus	No	Total upstream delay from the CM timestamp reference point to the CMTS timestamp reference point.
Trtt	Yes	Trtt = Tds + Tus. This is a measured value.
Ti	Yes	Total interleaver delay in the downstream path. The delay is equally shared between the CMTS and CM implementation.
Tds-cmts	Yes	Delay from CMTS timestamp reference point to CMTS output. This does not include the interleaver delay.
Tds-hfc	No	Delay of the HFC plant in the downstream.
Tds-cm	Yes	Delay from the input port of the CM to the CM timestamp reference point. This does not include the interleaver delay.
Tus-cm	Yes	Delay from the CM timestamp reference point to the CM output.
Tus-hfc	No	Delay of the HFC plant in the upstream.
Tus-cmts	Yes	Delay from the CMTS input port to the CMTS timestamp reference point. This delay should take into account the difference from where the CM timestamp was inserted into the upstream packet and the reference point used by the CMTS timestamp that the CMTS US PHY attaches to the packet.
A	Yes	An assigned variable that expresses the upstream to downstream asymmetry. This does not include the downstream interleaver delay or the upstream queuing delay or scheduler uncertainty. Asymmetry may come from differences in propagation delay at different frequencies and if there are any differences in path length between the downstream and upstream paths.

Table 3 – System Delay Definitions

Server is connected to GPS, then the DTI Server can align the DOCSIS timestamp with the GPS Epoch. The GPS epoch is January 6, 1980.

PTP references time as the number of nanoseconds after the epoch of the beginning of the day of Jan 1, 1972. PTP can also have arbitrary epochs.

In practice, the DTP approach is to first compensate for the time offset between the DOCSIS timestamp at the CM and the DOCSIS timestamp at the CMTS. Then, the DOCSIS timestamp at the CM is transformed into a PTP timestamp.

There are three tasks to be accomplished at the CM:

1. The least significant bits of the PTP timestamp are derived from the DOCSIS timestamp (and potentially the fractional timestamp extension).
2. The most significant bits of the PTP timestamp are derived from a signaling message
3. The offset that represents the delay from the CMTS to the CM is measured, calculated, signaled and then applied to the timestamp.

The PTP timestamp is defined in [1588] as seconds and nanoseconds from the original chosen epoch (which can be PTP or Arbitrary). The first field is the seconds field and is 48 bits long. The second field is the nanoseconds field and is 32 bits long. The nanosecond field never exceeds 10^9. This means that the PTP timestamp is referenced to 1 ns with a possible further resolution down to 15 femtoseconds by using a correction field (NTP has a 232 picoseconds resolution).

The DOCSIS timestamp is defined in [DOCSIS DRFI] as a 32-bit binary counter that is clocked with the CMTS 10.24 MHz master clock. This means that the DOCSIS timestamp is referenced to 97.65624 ns.

Because SCDMA allows for up to 128 CMs to transmit in the same timeslot with 128 orthogonal codes, the CMs must be aligned to within a fraction of a timeslot to avoid packet corruption. To accomplish this, an additional 8-bit fractional field advertised with a TLV is used providing a resolution of 1/16384 or 0.3814 ns.

Usage of the fractional field enables higher resolution time to be represented in SCDMA. PTP has the ability to express timing accuracy in fractional nanosecond resolution. It may be useful to include this fractional time field in the DOCSIS protocol or DTP extensions to increase PTP accuracy, even when using ATDMA.

Note that within the CM electronics, the DOCSIS timestamp is in a different clock domain than where the PTP timestamp is generated. The timestamp value must be transferred across the CM internal boundary in a consistent manner across multiple implementations.

Due to the limited size of the DOCSIS timestamp, it rolls over to zero approximately every 7 minutes. This means that the upper bits of the PTP timestamp should be sent more frequently than 7 minutes and that the CM mechanism must deal with the rollover when attaching the upper bits.

There are various ways of construction signaling. In one method, the various system offsets are measured and collected by the CMTS. The CMTS then sends a final correction value to the CM. In another method, the CMTS would publish any offsets it has, the CM would measure its internal offsets and then perform the final math.

The derivation of this offset is described in the next section.

TIME OFFSET TECHNIQUE

DOCSIS Path Latencies

The round trip DOCSIS path delay is inherently asymmetrical and can contain a moderate to high amount of jitter. Asymmetry and jitter introduce error into any timing protocol that might traverse the DOCSIS network. DTP mitigates these two factors by modifying the DOCSIS hardware design and deriving timing information directly from the DOCSIS system at the CM for use by NTP/PTP.

The packet transport delay in the downstream path of DOCSIS is relatively stable. It has a variety of fixed delays in the equipment and some variable propagation delay on the plant depending upon wind and temperature. The downstream interleaver delay is a programmable value and for DOCSIS is typically 0.68 ms (for 256-QAM). The length of the DOCSIS plant can be from zero to 100 miles. As a result, the one-way transit delay of the HFC plant can be up to 800 µsec. The PHY delays are unknown. The actual time that a bit passes through the external RF interface is indeterminate. Transit time will also depend upon other configuration parameters such as modulation order and FEC type.

It is not necessary to launch a separate signaling message with a timestamp in it in the downstream. DOCSIS already has a SYNC MAC Management containing a timestamp that is delivered from the CMTS to the CM with less than 500 ns of jitter, as specified in Section 6.3.9 of [DOCSIS DRFI]. The SYNC message bypasses the downstream output queues and their associated jitter and latency. The CM synchronizes itself to the timestamp in the SYNC message. The DOCSIS system will have to ensure that the jitter from the DOCSIS timestamp is sufficiently filtered so that it does not contribute error to the PTP timestamp.

The upstream DOCSIS path has much more uncertainty. To send a packet upstream, the CM must launch a request packet in a contention slot. If that fails, it keeps trying at longer and longer time intervals until it gets through. The CMTS then schedules a data transmission slot and issues a MAP MAC Management message. MAP messages tell CMs when to start and stop upstream transmissions, and what modulation profile to use.

This mechanism is actually quite efficient, but is not very predictable. This is one reason why timing protocols that are run over the top layer may see large variation in their results.

There are other scheduling techniques in DOCSIS, such as unsolicited grant service (UGS) or real time polling (RTP) services that can make the transmission opportunities more predictable. However, due to the natural jitter (on the order of 1 ms) in CMTS scheduling resolution, these alternative scheduling techniques do not provide enough accuracy to provide precision timing.

DTP relies on the DOCSIS system to take a series of measurements and then to supply the appropriate correction factor to the CM timestamp to arrive at the PTP timestamp.

DOCSIS Ranging

DTP relies partly on the DOCSIS ranging mechanism, so it is important to describe how that works.

It is important to realize that the CM receives the CMTS timestamp and then uses it directly as the CM timestamp. Thus, the CM timestamp is delayed with respect to the CMTS timestamp. In fact, the entire delay chain of the downstream, including portions of the CMTS, the HFC plant and the CM, contributes to the delay of the CM timestamp.

If the CM used this timestamp to transmit an upstream packet, that packet would arrive late at the CMTS. In fact, it would arrive late by an amount approximately equal to the entire downstream delay and the entire upstream delay.

To solve this problem, the CMTS has a two-part process. The first part is known as initial ranging and the second and re-occurring part is periodic maintenance.

The CMTS sets up a large (usually 2 ms) upstream Initial Ranging window. This window is contention based and any unregistered CM can attempt to register.

The CM sends a ranging request packet. The CMTS measures the arrival time error and sends back the error in a ranging response message. This continues until the system is working with specification limits.

The process is then repeated every 30 seconds or less. This is called periodic maintenance and is unicast in nature since the address of the CM is now known.

The net result is that the CM will calculate a ranging offset. It will then subtract this ranging offset from its timestamp to figure out when it really needs to transmit a packet.

There are two specific characteristics to note. The first is that the CMTS is not formally told what the ranging value is. The second is the ranging value held in the internal CM register can be unique to each implementation. There are many delay elements in the CM upstream design. The ranging offset is just one of them.

To recap, when the CM is told to transmit a packet to the CMTS at a particular time, it must send it earlier. To figure out how much earlier, the CM goes through a ranging process with the CMTS to create a ranging offset. It uses this ranging offset to transmit a packet earlier than the timestamp indicated in the DOCSIS MAP message. If ranging was done correctly, then the packet will arrive at the CMTS in the correct timeslot that it is supposed to.

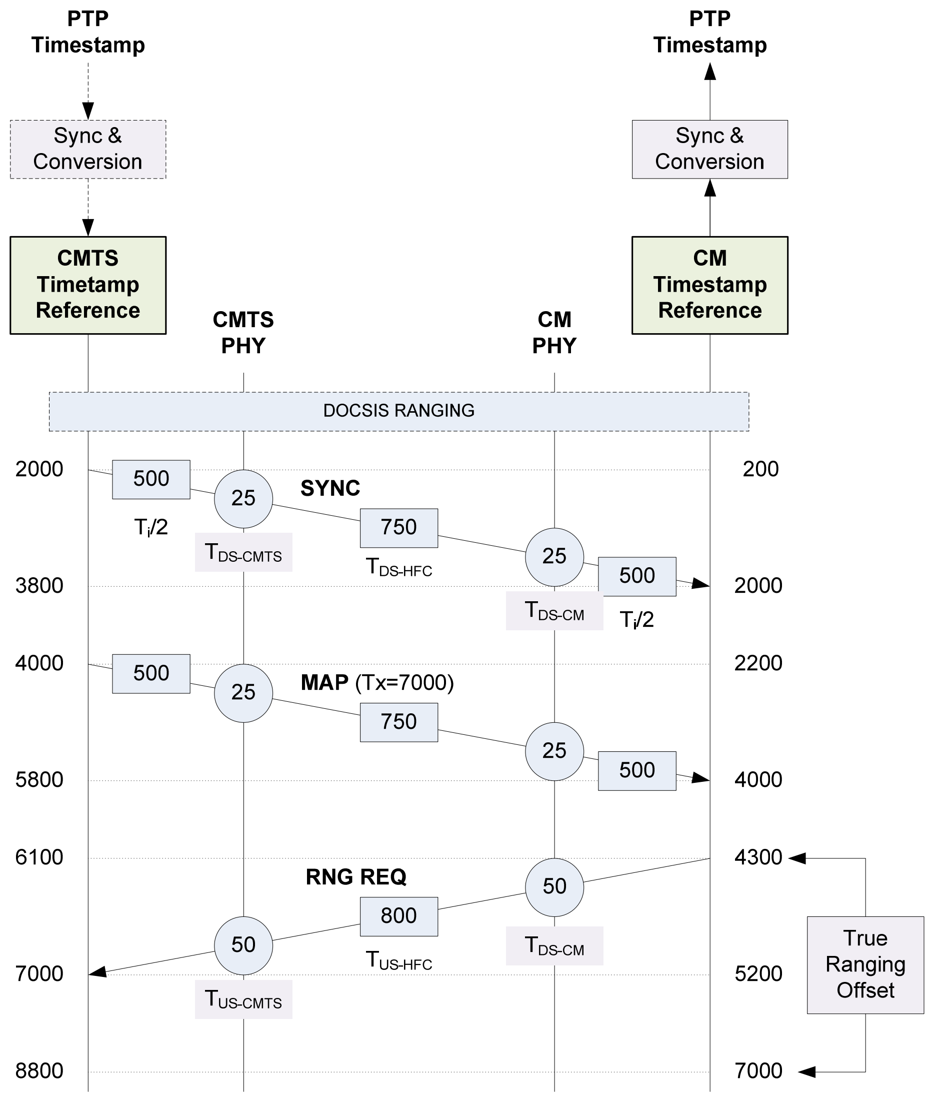
Measuring Round Trip Delay

The first DTP system measurement is a round trip delay. There are several different ways that this could be done. One way is to leverage the ranging process.

Since the actual ranging offset used in a CM is implementation specific, DTP makes a measurement. That measurement in DTP is called the true ranging offset (TRO).

That measurement is taken between the two reference points that matter to DTP – the CMTS timestamp, as referenced in the MAP message, and the CM timestamp.

The TRO of the CM in DTP is defined as the difference between the time the first bit of a packet is transmitted in the upstream from the CM in terms of the CM timestamp and the time the first bit of the packet is expected to arrive at the CMTS.



Where:
- All values are in arbitrary time units for sake of example.
- Upstream HFC delay is set to slightly more than downstream HFC delay
- DOCSIS ranging process determines internal offset for upstream tx time.

Thus:
Round Trip Delay (calculated) = 500 + 25 + 750 + 25 + 500 + 50 + 800 + 50 = 2700
True Ranging Offset (measured) = 7000 – 4300 = 2700

Formula Results:

Actual Offset Needed:	PTP Offset = 500 + 25 + 750 + 25 + 500 = 1800
Formula (1) approximation:	PTP Offset = (2700 – 1000) / 2 + 1000 = 1850
Formula (9) with Tus-off = 0:	PTP Offset = 1050 +(2700 – 1150) / 2 = 1825
Formula (9) with Tus-off = 50:	PTP Offset = 1050 +(2700 – 1150 - 50) / 2 = 1800

Figure 7 – True Ranging Offset

The expected packet arrival time at the CMTS is listed in the MAP message. So, the CM has to store that value, capture the local CM timestamp value when the correct packet transmits, and subtract the two.

In essence, *the total round trip delay is equal to the true ranging offset of the CM.*

The ranging offset of the CM is intended to correct for the round trip delays in the DOCSIS system. Thus, it should be possible to measure the operating state of the CM and reverse engineer what those network delay values are.

An Example

An example of this process is shown in Figure 7. Somewhat arbitrary values are used for illustration purposes only. The PHY delays were intentionally set different, and the upstream was given more delay than the downstream.

This example shows how the CMTS can either receive network timing or be self-contained for timing. The CM timestamp is synchronized to the DTP timing source and converted to a PTP timestamp.

DOCSIS Ranging occurs. By a process of trial and measurement, the CM arrives at a ranging offset that works for its particular implementation.

When the SYNC message traverses from the CMTS to the CM, the CM uses the value of the timestamp it receives. In this example, a value of 2000 was sent and received. Due to the downstream delay, when the CM timestamp is at 2000, the CMTS timestamp has already advanced to 3800.

Next a MAP is received that tells the CM to transmit a packet at the time of 7000. Using its ranging offset, the CM launches this packet when the CM timestamp is 4300. Because of the ranging process, the CMTS receives the packet at the time of 7000.

The true ranging offset can be measured after the ranging process is complete by taking the difference of the timestamp in the MAP and the timestamp in the CM corresponding to the start time of upstream transmission in the MAP. Note that the true ranging offset will generally have a fixed offset from the actual ranging offset used in the CM.

It can be seen that the ranging circuitry of the CM picked an actual ranging offset that caused the true ranging offset to be equal to the round trip delay.

Caveats

The true ranging value is a offset that can be measured by the CM and reported to the CMTS. Note that this value may not exactly equal the actual ranging value used in current implementations, since there are other circuit delays involved in the use of this value.

Any portion of the round trip that is outside of the measurement path cannot be included in the measurement. However, if it can be defined, a correction factor can be applied. For example, if there is a delay in the CMTS between the receive timestamp and the transmit timestamp, the CMTS will have to provide a correction factor.

In theory, the true ranging offset could change with every ranging interval. Ranging intervals occur every 25 to 30 seconds. Such changes can occur if the delay of the plant increases due to temperature shifts (reference Appendix VIII of [DOCSIS 2.0]. As a result, the CM may choose to time average the true ranging offset a finite amount of time to remove this uncertainty. Too long a period of time should be avoided as it would impact the CMs ability to react to network changes that would then impact the value of the PTP timestamp at the CM.

The timestamp is also being constantly updated with the SYNC message at least every 200 ms. Even though the CMTS and CM are frequency locked through the downstream PHY, if there was enough of a change in the delay of the downstream path, the timestamp value at the CM would be adjusted over time to the new value from the CMTS. This will also impact the PTP timestamp value.

The true ranging offset can only changed during periodic ranging. The natural packet to use for upstream measurements is then the DOCSIS MAC Ranging Request message.

Alternatively, other approaches could be used that focused on a different MAC management message. For best results, the upstream packet should be contained within a single carrier (no bonding). Fragmentation, concatenation, and CCF (continuous concatenation and fragmentation) should be disabled. These requirements are met with the Ranging Response message.

Sometimes, a DOCSIS system uses a different upstream PHY profile for different upstream operations (ranging vs. data for example) requests. A different modulation profile could result in a different upstream path delay. This may be okay for this particular application. Further analysis is needed. However, if the upstream delay is needed for other applications, this mechanism may need a correction factor, a different upstream message with which the measurement is made, or a ranging packet with the same PHY profile as the upstream data path.

First Pass Approximation

At this point, the round trip delay is now known and an approximation can be made of the time offset needed for PTP. However, there is still some information missing. The delay through the PHY circuitry at the CMTS and CM transmit and receive is not known. Further, the asymmetry of the downstream path and upstream path is not known.

The approximation would be to subtract out the downstream interleaver delay, assume all four PHY delays are symmetrical, and that the remaining downstream and upstream DOCSIS paths are symmetrical. Then divide the measured path by 2.

$$Approximate\ Offset = (T_{rtt} - T_i) / 2 + T_i \quad (1)$$

But, what if this is not accurate enough? If we could determine the total asymmetry of the DOCSIS path, then a more accurate offset could be calculated. The next step is to derive the one-way delay in the downstream.

Measuring DOCSIS Asymmetry

Even better accuracy can be achieved if the CMTS has access to a reference CM that is identical in build (same manufacturer and same model) to the CM in the field. It can then compare measurements on the reference CM to the remote CM.

If there is more than one type of CM deployed that shall provide precise time downstream, there may have to be more than one local reference CM. If there is more than one type of CMTS line card, then there may have to be a duplicate reference CM on each unique CMTS line card.

The CMTS will program the reference CM with the same PHY configuration as the remote CM. The same software should be loaded as well (although that is generally not under the control of the CMTS). The DOCSIS system then performs two measurements.

1. It makes a round trip measurement.
2. It makes a one-way downstream path delay measurement.

The downstream path delay measurement is made by connecting the PTP or 1PPS output port of the reference CM into a PTP slave input port or a 1PPS input port on the DTI Server or on the CMTS.

If the external DTI Server is used, then it measures the delta between the PTP timestamp and the DOCSIS timestamp and reports it over the DTI interface to the CMTS.

If the reference CM has the right offset, then the timestamp delta will be zero (PTP timestamp is converted to a DOCSIS timestamp to perform the math) and the total downstream delay will be represented by the PTP offset used by reference CM.

One approach is to adjust the PTP timestamp offset of the local CM until the error between the local CM output and the CMTS timestamp, as measured externally, is minimal.

Offset Math

There are several ways to put the numbers together. Further, the calculations could be done at the CMTS or CM that will impact the approach slightly. Here is one basic method.

The system diagram for this example is in Figure 6. The definition of the variables is in Table 3.

The downstream delay for the reference CM is a measured value with a near-zero (and therefore ignored) HFC plant path length, and is defined as follows:

$$Tds\text{-}ref = Ti + Tds\text{-}cmts + Tds\text{-}cm \quad (2)$$

The downstream delay for the remote CM differs by the path length of the HFC plant downstream.

$$Tds = Tds\text{-}ref + Tds\text{-}hfc \quad (3)$$

The round trip time for the reference CM is a measured value with a near-zero HFC plant path length, and is defined as follows:

$$Trtt\text{-}ref = Ti + Tds\text{-}cmts + Tds\text{-}cm + Tus\text{-}cm + Tus\text{-}cmts \quad (4)$$

The round trip time for the target CM differs by the path length of the HFC plant downstream.

$$Trtt = Trtt\text{-}ref + Tds\text{-}hfc + Tus\text{-}hfc \quad (5)$$

Let’s assign linear correction factor to the HFC plant asymmetry called Tus-off. Tus-off expresses the additional amount of the upstream delay when compared to the downstream delay. Tus-off would be assigned based upon the operator’s knowledge and characterization of the plant. For example, Tus-off could account for group delay differences between the DOCSIS downstream and upstream carrier frequencies. Note that Tus-off does not represent any asymmetry within the hardware of the CMTS or the CM itself since the reference cable modem removes that asymmetry.

For example, if Tus-off = 50 ns, then the upstream path would have 50 ns more latency than the downstream path.

$$Tus\text{-}off = Tus\text{-}hfc - Tds\text{-}hfc \qquad (6)$$

$$Tus\text{-}hfc = Tds\text{-}hfc + Tus\text{-}off \qquad (7)$$

Applying equation (7) to (5) and solving for the downstream hfc delay,

$$Trtt = Trtt\text{-}ref + Tds\text{-}hfc + Tds\text{-}hfc + Tus\text{-}off$$

$$Tds\text{-}hfc = (Trtt - Trtt\text{-}ref - Tus\text{-}off) / 2 \quad (8)$$

Applying equation (8) to (3) yields the final equation for the offset of the downstream timestamp.

$$Tds = Tds\text{-}ref + (Trtt - Trtt\text{-}ref - Tus\text{-}off) / 2 \quad \ldots (9)$$

Applying formula (9) to the example in Figure 7 with no HFC correction factor (Tus-off = 0 ns) and where the plant length is 0 for the reference CM yields:

Tds = 1050 + (2700 – 1150) / 2 = 1825

and correcting for HFC plant asymmetry using the value from the example for Tus-off = 800 – 750 = 50,

Tds = 1050 + (2700 – 1150 – 50) / 2 = 1800

As an alternative calculation, the asymmetry could be expressed as a ratio of Tds-hfc and Tus-hfc.

What about DPV?

DOCSIS 3.0 has a MAC management message called DOCSIS Path Verify (DPV). DPV allows the beginning and ending timestamp in each direction of the link to be captured and analyzed by the CMTS.

DPV has similar goals but less accuracy than the measurement technique discussed in DTP. DPV packets will see queuing delays in the upstream path where as the current DTP proposal does not. This is because DPV is using a timestamp (reference point U_1) generated prior to queuing. Refer to Section 10.5.2 "DPV Reference Points" in [DOCSIS MACUP].

As an alternative implementation of DTP, DPV could be improved if the timestamps were provided directly by a hardware mechanism after packet queuing and just prior to transmission (theoretical reference point U_1').

Measuring the true ranging offset may be an easier implementation for a CM design than modifying an upstream DPV packet.

DTI Server Recap

Here is a recap of the system requirements from the point of view of the DTI Server.

The DTI existing functionality supports the generation, maintenance and distribution of precision time. The DTI server function shall be extended to support the precision PTP monitoring function. The PTP monitoring function is that portion of the DTI server that measures any timing offset between the reference CM and the CMTS.

The precision PTP monitoring function includes the following elements:

1. The PTP monitoring function provisioning is controlled externally. The control function will reside in the CMTS that also manages the pool of co-located reference CMs.
2. The PTP monitoring function supports establishment of monitoring sessions. In a monitoring session the DTI server shall be operated as an ordinary client in the 1588 protocol exchange.
3. The PTP monitoring function shall collect timestamp data for each session and extract an estimate of time alignment error with respect the DTI server precise 1PPS reference.
4. The PTP monitoring function shall perform the time error estimation task based on a schedule. The schedule of the start and duration of each measuring session is provided by the external control function.

5. The DTI PTP monitoring function shall support a minimum of eight simultaneous sessions.
6. The DTI PTP monitoring function may support a calibration function to mitigate asymmetry in the Ethernet connection between the co-located reference cable modem and the DTI server. In this mode one physical Ethernet port on the DTI server shall operate as a master and the Ethernet cable that normally terminates on the reference CM will be temporarily connected to the calibration master port. The calibration session control shall be supported in the DTI server using existing SNMP and CLI user controls.
7. The PTP monitoring function required a minimum of two physical Ethernet ports to support the calibration function. Optionally, supporting one port per simultaneous session will provide the highest achievable measurement accuracy.

ADDITIONAL SOURCES OF ERROR

There are a series of minor errors that either can be ignored or compensated for.

Reference CM Precise Timing Output

There is a delay from the reference CM Ethernet Master Clock output to the adjacent DTI Server Slave Clock input. Using the PTP delay_req and delay_resp messages, the symmetrical delay between the reference CM and the DTI server can be automatically removed by PTP. If using a 1PPS instead of PTP to connect the reference CM to the DTI Server the cable length could be pre-configured by the user to remove the delay.

DTI Server Propagation Delays

The DTI Server must make a difference measurement between the Reference CM and the CMTS Timestamp. If there are any offsets in the DTI circuit, it must compensate for them.

The DTI Server could be separate or embedded in the CMTS. Alternatively, the CMTS could host a PTP or 1PPS input and do the delta measurement with its own timestamp, even if there is an external DTI server.

Differences in CM Hardware & Software

To minimize measurement error, the two CMs under measure should be identical models from the same manufacturer and be running the same firmware and configuration. Under certain circumstances, it may be necessary to have a common manufacturing lot number to ensure the same performance in items such as tuners or LFE which are external to the CM silicon.

Differences in CMTS Hardware & Software

In the ideal case, the same CMTS linecard should be used for the remote CM and the reference CM. If the reference CM is on a different line card, then any differences between the two line cards can contribute to error.

This may not be practical if there are many remote modems all running PTP. At the very least, there should be one local reference CM per type of CMTS line card used.

Ranging Accuracy

A CM is ranged to a particular degree of accuracy. This should be analyzed to see if the residual ranging error, if any, would impact the accuracy of the PTP timestamp.

Upstream Interleaver

There is an interleaver that operates at the packet or burst level in the upstream direction [DOCSIS DRFI]. If the interleaver is implemented prior to packet queuing at the CM and after the packet queuing at the CMTS, then it is outside the transmission path and can be ignored. If it is created in real time and is part of the transmission path, then it could be included in asymmetry calculations.

METHODOLOGY

DOCSIS has the same goal as PON and DSL in that they all provide an access network technology that can be used for the last mile connectivity to end user. All of these last mile technologies use aggregation networks that tend to be based on IEEE 802.3 Ethernet technology.

PON, DSL links, and DOCSIS over cable plants introduce large PDV and asymmetry that are detrimental to packet-based timing distribution. This issue has been recognized and in the ITU-T, relevant Questions 2 and 4 in Working Group 15 have worked on developing their own time distribution mechanism.

By using a method targeted at the access network technology, the time transfer can be optimized, leveraging information specific to the access technology. As such, PDV and asymmetry can be reduced or compensated for, allowing better time transfer than simple transmission of PTP messages over the access network.

Such specific time distribution methods cannot be used over other media. For example, it is very difficult to provide precise timing with PTP over the top of a DOCSIS network to Ethernet based equipment located beyond a cable modem. Moreover, if the timing source is not co-located with the line termination equipment, such as a CMTS, the time shall be transferred to the DTI server or to the CMTS by other means.

At both ends of the access network, at least one other mechanism such as IEEE 1588 PTP or NTP would have to be used with some adaptation layer to transfer the time between the two network sections. If the continuity of the time signal can be maintained, the traceability of the time from the source may be broken without a specific adaptation function. This paper introduces such solution, named DOCSIS Timing Protocol (DTP).

In addition, the access network, that is, the pair PON OLT/ONU, DSLAM/DSL modem or CMTS/CM, through the utilization of their specific time distribution (e.g., refer to [GPON], [XG-PON] or clause 13 of [802.1AS] for EPON; for other access networks such as VDSL2, work is still in progress), may assist other packet-based timing protocol such as NTP or PTP.

In cases when timing packets must pass through the access link, the access network devices may simulate or act as virtual or distributed IEEE 1588 boundary clock, NTP stratum server or IEEE 1588 transparent clock. In such a design model, the PTP or NTP communication path would not be broken but the access network would provide correction for the PDV and asymmetry.

Hence multiple design scenarios based on DTP can be evaluated.

DEPLOYMENT SCENARIOS

Some customers, such as residential customers (e.g., for home femtocell), may just need a high quality timing source without being concerned about its traceability.

Other customers, such as power utility companies, mobile or broadcast operators may need information about the timing source in order to correlate time between distinct end devices that must be synchronized. This level of information may be requested for traceability or accountability.

Moreover, the operator providing a timing service to its customers may also have its own constraints that drive them to distinct timing distribution architecture.

The following figures depict three main deployment scenarios with some variants (a, b…). These are summarized in Table 4.

Those scenarios provide distinct timing infrastructures from the end user's and/or operator's viewpoint and offer flexibility to the timing infrastructure. However, each scenario may impose its specific requirements and configurations at the CMTS and CM.

#	Description
1	Timing source at DTI Server/CMTS location
1a	CM acts as IEEE 1588 grandmaster (or NTP server)
1b	CM acts as IEEE 1588 grandmaster faking the DTI Server/CMTS
1c	DTI Server/CMTS acts as IEEE 1588 grandmaster (or NTP Stratum 1 server) CM acts as IEEE 1588 boundary clock (or NTP stratum server)
2	Timing source and IEEE 1588 grandmaster (or NTP server) is upwards the DTI Server/CMTS location
2a	DTI Server/CMTS fakes the IEEE 1588 grandmaster (or NTP Stratum 1 server)
2b	DTI Server/CMTS/CM acts as a distributed IEEE 1588 boundary clock (or NTP servers)
2c	DTI Server/CMTS and CM are virtual IEEE 1588 boundary clocks (or NTP servers)
2d	CMTS and CM are distributed or virtual IEEE 1588 boundary clocks (or NTP servers); DTI server is removed from timing communication path
3	CMTS/CM acts as a distributed IEEE 1588 transparent clock

Table 4 – Possible Scenarios of Timing Transfer

The nomenclature used for the following diagrams is shown in Figure 8:

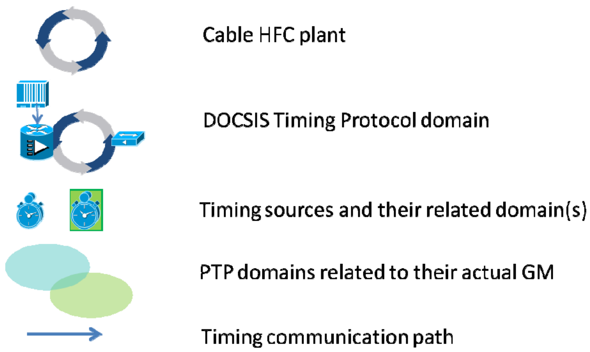


Figure 8 - Scenario Nomenclature

Scenario 1

Scenario 1a represents the baseline configuration.

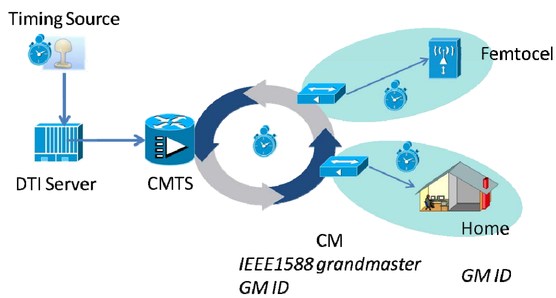


Figure 9 - Scenario 1a

The CMTS is provided precise timing by the DTI server. CM acts as a IEEE 1588 grandmaster (GM) using timing sourced by the DTI server/CMTS via the DTP method. The DTI server/CMTS shall also provide some timing source information so that the CM's GM function can populate the PTP dataset members transmitted to the downstream PTP slaves. The CM would provide its own clockID for grandmaster identifier (GM ID).

Utilizing the same DTP distribution variants 1b and 1c allow the PTP clocks beyond CMs to trace the real-timing source and identify the DTI server/CMTS as GM. In such scenarios, the CMs would have to use a DTI server/CMTS PTP clockID for the GM ID.

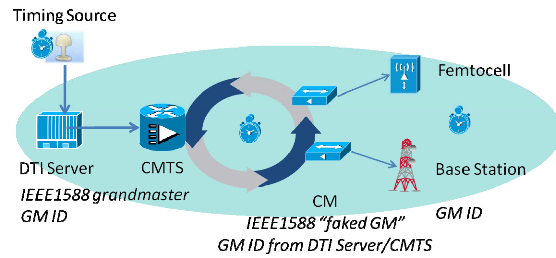


Figure 10 - Scenario 1b

In scenario 1b, the CM remains the real IEEE 1588 GM but replaces its own clockID by the DTI server/CMTS ClockID.

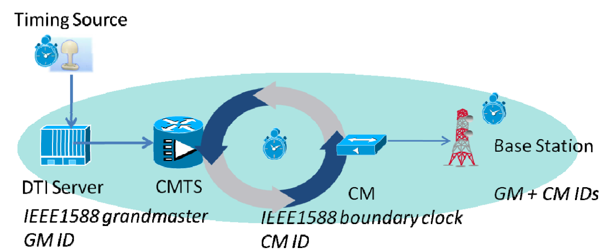


Figure 11 - Scenario 1c

In scenario 1c, despite the fact that there are no PTP messages being sent over the cable path to the CM, the CM simulates a boundary clock. It generates PTP messages with its own clockID but uses the DTI server/CMTS PTP clockID for GM ID. As a result, all CMs connected to this DTI server/CMTS would be related to same "GM".

Scenario 1c may be useful for providing PTP traceability to the "GM" by providing the distinct IDs. In contrast, in scenario 1b, the CM would substitute for or fake the DTI server/CMTS as GM.

For those scenarios, the CMs can be managed by PTP management or through

MIBs. The DTI server/CMTS can be a PTP management node or a Node Manager MIB with MIB extensions for the CMs acting as grandmaster or boundary clock as depicted in Figure 12.

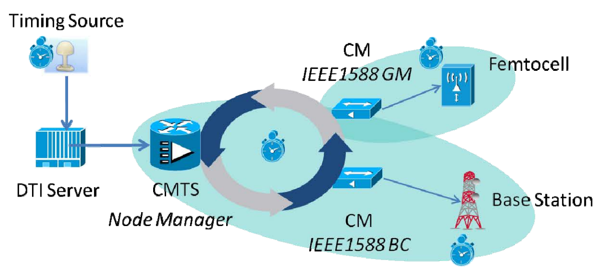


Figure 12 - Scenario 1 Management

Scenario 2

When the primary or a backup timing source and IEEE 1588 GM are remote to the DTI Server/CMTS locations (for instance somewhere in the aggregation or core network), other scenarios are conceivable.

From a timing domain viewpoint, the main difference would be to have the same source/GM for multiple customer clocks. Those clocks might be spread over multiple CMTSs, HFC plants and, also accessible from other access networks (e.g., Ethernet).

In such scenarios, we have to consider the transmission of the timing references towards the DTI Server/CMTS location then to the PTP clocks beyond the CMs. As shown in Figure 13, the DTI server and CMTS have no local timing source. The timing source (e.g., GPS) and GM are remote.

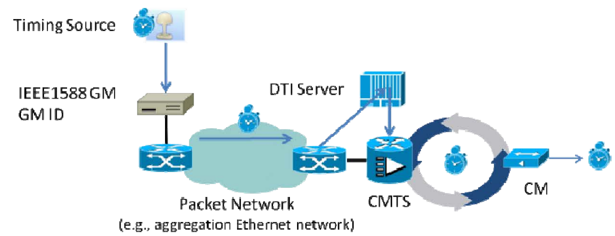


Figure 13 - Scenario 2

Transmission of timing signals from reference(s) over a packet network towards the cable plants can be achieved by the normal TWTT protocol with network assistance as described earlier (e.g., Synchronous Ethernet for physical-layer frequency, hardware assistance to PTP or NTP packets).

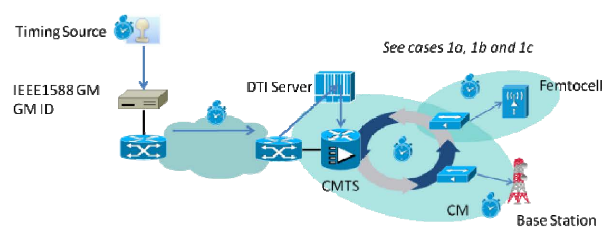


Figure 14 - Scenario 2a

In scenario 2a, the DTI Server/CMTS/CM conceptually can behave as in any previous scenarios 1a to 1c. The difference is that the DTI server will use the network timing signal(s) instead of a local timing source. This scenario does not provide traceability to the actual timing source and IEEE 1588 GM.

Hiding or removing the traceability to the central GM may be intentional, for instance, for timing and management domain delimitation/creation towards the customer network. The information at the CM is defined at the DTI server/CMTS. But because the information sent by the CMTS to the CM comes from the GM via DTI, the operator can trace back to the real GM.

Conceptually the next scenarios may look like one distributed boundary clock (scenario 2b) or two virtual boundary clocks (scenario 2c).

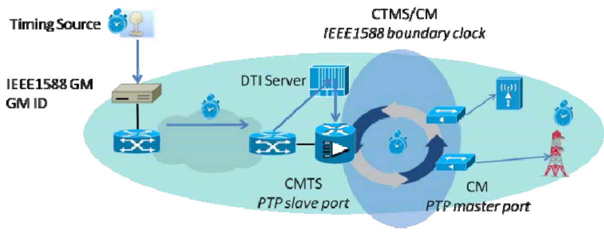


Figure 15 - Scenario 2b

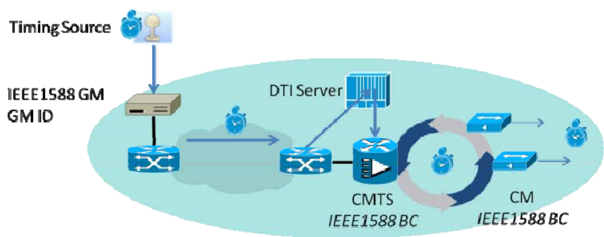


Figure 16 - Scenario 2c

The DTI/CMTS recovers the time from central GM, i.e., the DOCSIS time is synchronized to central timing source. The DTI Server/CMTS and the CM must modify the information from the central GM before being delivered to clocks beyond the CM via PTP.

Similar to scenarios 1b and 1c, the customer clocks can trace back to the real central timing source and GM, not the DTI server/CMTS fake GM.

Differences between the one BC and two BCs scenarios come from the clockID being used for traceability and from the count of PTP hops. In case of one BC scenario, the clock ID may be the clockID of the DTI Server/CMTS, the one from the CM or a distinct clockID that would represent the BC.

The BC hop count (PTP "removeStep" dataset member) would be incremented by one. For a two BC scenario, the DSIT Server/CMSTS and CM would use their respective clockID and the BC hop count would have to be incremented by two. This is again a distinction related to PTP communication path management and monitoring.

Note that in scenario 2c, the CMTS and CM are not actual IEEE 1588 BC because they do not receive PTP messages. We might consider that the CMTS may receive directly the PTP messages and recover time (PTP slave port) without going through the DTI server and necessary associated manipulations. This is depicted by Scenario 2d.

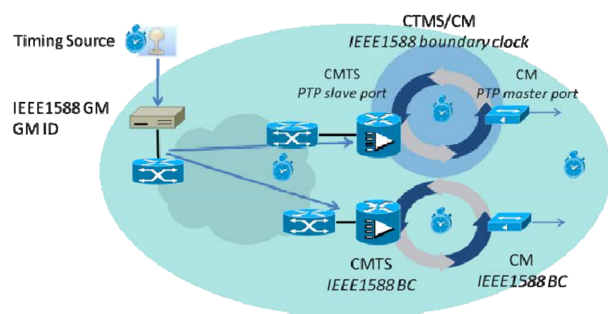


Figure 17 - Scenario 2d

Scenario 3

A final alternative would allow the HFC plant to be independent of the timing source located in the network.

Conceptually this scenario may look like a distributed transparent clock.

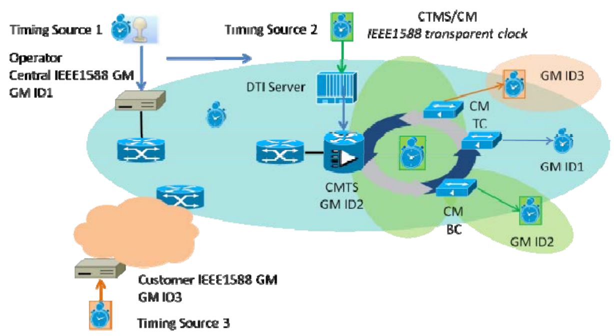


Figure 18 - Scenario 3

In this scenario the DTI Server and DTP can use distinct timing references. This scenario allows timing source 2 to be transmitted to CM as depicted by scenarios 1a to 1c. However, the PDV correction provided by DTP could be a benefit to the PTP traffic sent "over" the cable path just as an IEEE 1588 transparent clock would do. Such a mode would permit correcting any time reference (such as timing source 1) distributed over the cable plant, allowing the support of multiple timescales.

From PTP management viewpoint, the CMTS can still play as node manager for the connected CMs. A more centralized approach can also apply.

A mix of solutions may be useful because different application, customer and timing domain management. However only a subset of the presented various scenarios should be considered.

SUMMARY OF DESIGN CHANGES

The authors of this paper will be pursuing these proposed changes with the appropriate standards committees and interested vendors.

CM

Hardware

- MUST be able to lock to the DOCSIS downstream baud clock, regardless of the upstream modulation type.
- MUST filter jitter from the DOCSIS downstream baud clock to a level acceptable for an Ethernet clock.
- MUST support Synchronous Ethernet.
- MUST have a PTP output circuit on its Ethernet port that inserts a PTP timestamp.
- MUST be able to measure the difference between the transmit time of an upstream packet in terms of the local CM timestamp and the CMTS timestamp in the MAP.
- MAY have a 1PPS output.

Software

- MUST have a PTP Stack
- MUST support DOCSIS protocol changes for DTP.

CMTS

Hardware

- MAY have a PTP slave input for network sync.
- MAY have a PTP slave input and/or a 1PPS input for connecting to reference timing CMs.
- MUST synchronize downstream baud clock to CMTS master clock.

Software

- MUST support DOCSIS protocol changes for DTP

- MAY support DTI protocol changes for DTP
- MUST support the coordination of DTI Server and CM operations for DTP.

DTI Server

Hardware

- MUST have a PTP Slave port for connectivity to CM under measurement.
- MUST have a PTP Slave port for network sync connectivity. These two slave ports MAY share a common physical port.
- MAY have a 1PPS input.
- MUST have the ability to compare the timestamp from the PTP slave port when received directly to the DOCSIS timestamp.

Software

- MUST support DTI protocol changes for DTP.

DOCSIS Protocol Changes

DS MSG:

- Upper bit PTP prefix (for operation greater than 7 minutes)
- Lower bit PTP suffix (for resolution down to 1 ns) based upon the fraction time field originally intended for SCDMA-only use.
- Translation of DOCSIS timestamp with suffix and prefix to EPOCH
- Clock ID

US MSG

- True ranging offset.

DTI Protocol Changes

CMTS to DTI Server

- DTP measurement request

DTI Server to CMTS

- DTP measurement response

CONCLUSION

The DOCSIS system is already based upon highly precise timing. Rather than running timing protocols such as NTP and PTP independently over-the-top of DOCSIS, better performance can be achieved by leveraging the precision timing already native to the DOCSIS system that allows the CM to be time and frequency synchronized to the CMTS.

Using this as the cornerstone, the time and frequency synchronization of the CMTS and CM can be used to correct any path delay variation through the network when acting as a transparent clock thus allowing more effective PTP over-the-top of DOCSIS. This approach would even allow the support of multiple timing domains.

Similarly, the DTP system can accurately generate time and frequency from the CM to natively support different protocols and interfaces like PTP, NTP, Synchronous Ethernet or 1PPS.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the contributions of George Zampetti, the Chief Scientist of Symmetricom and the primary author of the DTI Specification [DTI] in the proving out of this concept and in the review of this white paper. The authors would also like to thank Michael Overcash of Cisco for his contributions.

REFERENCES

[1588]
IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, IEEE Std 1588-2008, http://standards.ieee.org/findstds/standard/1588-2008.html

[1588 Applications]
Recent Advances in IEEE 1588 technology and Its Applications, John C. Eidson, July 19, 2005

[1588 Tutorial]
IEEE 1588 Tutorial, Prof. Hans Weibel, October 2, 2006

[802.1AS]
IEEE 802.1AS-2011 Timing and Synchronization for Time-Sensitive Applications in Bridged Local Area Networks

[802.3bf]
IEEE P802.3bf TimeSync Task Force, 2011 http://www.ieee802.org/3/bf/public/index.html

[G.8261]
ITU-T G.8261-Y1361-200804, Timing and Synchronization Aspects in Packet Networks (and related Corrigendum, Amendment and Addendum)

[G.8262]
ITU-T G.8262-Y.1362 (201007), Timing Characteristics of Synchronous Ethernet Equipment Slave Clock (EEC) (and related Corrigendum, Amendment and Addendum)

[G.8264]
ITU-T G.8264-Y.1364-200810, Timing Distribution Through Packet Networks, (and related Corrigendum, Amendment and Addendum)

[G.8271]
ITU-T SG15 Q13 work in progress recommendation which will define the Time and Phase synchronization requirements and network limits

[DOCSIS 2.0]
CM-SP-RFIv2.0-C01-081104, DOCSIS Radio Frequency Interface Specification, CableLabs, August 11, 2004

[DOCSIS DRFI]
CM-SP-DRFI-I11-110210, DOCSIS Downstream RF Interface Specification, CableLabs, issued Feb 10, 2011

[DOCSIS MACUP]
CM-SP-MULPIv3.0-I15-110210, DOCSIS 3.0 MAC and Upper Layer Protocols Interface Specification, CableLabs, Issued Feb 2, 2011

[DTI]
CM-SP-DTI-I05-081209, DOCSIS Modular Headend Architecture Timing Interface Specification, CableLabs, issued Aug 12, 2009, and ITU-T J.211-200611, Timing Interface for Cable Modem Termination Systems

[GPON]
ITU-T G.984.4-200911-Amd2

[XG-PON]
ITU-T G.987.3-201010 (Pre-published)