



## **Factory and Equipment Clock Synchronization and Time-Stamping Guidelines: Version 2.0**

**International SEMATECH Manufacturing Initiative  
Technology Transfer #06094781B-ENG**

**Advanced Materials Research Center, AMRC, International SEMATECH Manufacturing Initiative, and ISMI** are servicemarks of SEMATECH, Inc. **SEMATECH** and the **SEMATECH** logo are registered servicemarks of SEMATECH, Inc. All other servicemarks and trademarks are the property of their respective owners.

**Factory and Equipment Clock Synchronization and Time-Stamping  
Guidelines: Version 2.0  
Technology Transfer #06094781B-ENG  
International SEMATECH Manufacturing Initiative  
June 30, 20088**

**Abstract:** These guidelines from the MFGM031M project provide an overview of the components for quality clock synchronization. They complement the SEMI E148 time synchronization standard by recommending practices to implement SEMI requirements. When followed, these guidelines will tag data with accurate timestamps that enable data from multiple sources to be merged. This revision includes additional guidelines for equipment internal time synchronization and additional details about Network Time Protocol (NTP) commands used to verify time synchronization quality and performance.

**Keywords:** Computer Software, Data Management Standards, Standards Conformance

**Authors:** Harvey Wohlwend and Gino Crispieri (ISMI) and Ya-Shian Li (NIST)

**Approvals:** Harvey Wohlwend, Project Manager/Author  
Olaf Rothe, Program Manager  
Joe Draina, Director  
Laurie Modrey, Technology Transfer Team Leader



## Table of Contents

1	EXECUTIVE SUMMARY .....	1
2	BACKGROUND .....	1
3	REFERENCES .....	2
	3.1 Standard References .....	2
	3.2 Other References .....	2
4	TERMINOLOGY .....	3
5	KEY SEMICONDUCTOR INDUSTRY REQUIREMENTS AND RECOMMENDATIONS .....	5
	5.1 Clock Synchronization .....	5
	5.1.1 Requirements .....	5
	5.1.2 Recommendations .....	5
	5.2 Time-Stamping .....	6
	5.2.1 Requirements .....	6
	5.2.2 Recommendations .....	6
6	TIME SYNCHRONIZATION GUIDELINES .....	7
	6.1 Timescale .....	7
	6.2 Time Servers .....	8
	6.3 Traceability .....	9
	6.4 Fault Tolerance .....	10
	6.5 Time Source .....	10
	6.5.1 Global Navigation Satellite System Receivers .....	11
	6.5.2 Radio Receivers .....	11
	6.5.3 Internet .....	11
	6.5.4 Dial-Up .....	12
	6.6 Synchronization Protocol .....	12
	6.6.1 NTP .....	12
	6.6.2 PTP .....	13
	6.6.3 Hybrid (NTP and PTP) .....	14
	6.7 Client Configuration .....	16
	6.8 Synchronization Architecture .....	17
	6.8.1 Stratum Levels .....	21
	6.9 Clock Quality .....	22
	6.9.1 Precision Oscillators .....	23
	6.9.2 Quartz Oscillators .....	23
	6.10 Clock Resolution .....	24
	6.11 Synchronization Frequency .....	24
	6.12 Network Jitter .....	25
	6.13 Synchronization Performance .....	26
	6.13.1 Time Synchronization Monitoring Tools .....	27
	6.14 Security Practices .....	27
	6.14.1 Firewalls .....	28
	6.14.2 Access Control .....	28
	6.14.3 Authentication .....	28

6.15	Equipment Internal Time Synchronization .....	29
6.15.1	Robust .....	29
6.15.2	Resource Aware .....	30
6.15.3	Configurable.....	30
6.16	Data Time-Stamping Guidelines.....	30
6.16.1	Time Base .....	30
6.16.2	Format .....	31
6.16.3	Time-Stamping.....	31
6.16.4	Time-Stamping Resolution .....	32
APPENDIX A – ACCURACY/PRECISION GUIDELINES FOR e-MANUFACTURING APPLICATIONS .....		33

### List of Figures

Figure 1	Co-Existing NTP/PTP Time Synchronization Protocols.....	15
Figure 2	PTP-Based Time Synchronization With NTP Gateway .....	15
Figure 3	Client-Server Mode .....	18
Figure 4	Symmetric Active/Passive Mode .....	18
Figure 5	Broadcast/Multicast Mode .....	19
Figure 6	Manycast Mode .....	20
Figure 7	Example NTP Hierarchy for Factory Distributed Time Synchronization .....	21
Figure 8	Autokey Protocol Exchange.....	29

### List of Tables

Table 1	NTP/IEEE 1588 Comparison .....	14
Table 2	Clock Quality .....	23
Table A-1	On-Tool Application – Single Piece of Equipment Context .....	34
Table A-2	Close to the Equipment Applications – Group of Identical/Similar Tools.....	35
Table A-3	Factory/Process Equipment Application – Group Related Tools .....	35
Table A-4	Factory Level Application – Factory Logistics .....	36
Table A-5	Enterprise Level Application – Single Company .....	36

## 1 EXECUTIVE SUMMARY

Because of the multiple uses of time and the need to coordinate manufacturing processes, time synchronization is essential in distributed factory environments to ensure time-stamping quality and to rapidly diagnose process faults. To ensure the factory can readily synchronize an equipment internal clock and determine the quality of the time and timestamps generated by that clock, equipment suppliers should adhere to standards and techniques for time synchronization. These guidelines outline the components for quality clock synchronization and complement the SEMI time synchronization standard by recommending practices to implement the SEMI requirements. This revision includes additional guidelines for equipment internal time synchronization and details about Network Time Protocol (NTP) commands used to verify time synchronization quality and performance.

The audience expected to benefit from these guidelines includes standards developers, factory systems engineers, and application engineers from suppliers of equipment, software, subsystems, or internal custom applications.

## 2 BACKGROUND

Increasing wafer sizes, decreasing critical dimensions, and new materials all challenge yield, manufacturing costs, and time-to-market demands. Among the key issues driving the challenges are monitoring and maintaining surface uniformity, decreasing process tolerance windows, and increasing process and equipment complexity. Solving these issues will require data from process equipment, in-line metrology, and factory environmental conditions to be collected and analyzed. Managing the data proliferation to extrapolate intelligent conclusions in a timely manner will be critical in realizing the anticipated benefits of greater data accessibility enabled by Interface A (i.e., Equipment Data Acquisition [EDA]).

Accurate clock synchronization and timestamps are fundamental to achieving e-Manufacturing paradigms requiring real-time control such as advanced process control (APC) and e-Diagnostics, as well as development of an intelligent automated knowledge base for rapid yield learning. Time synchronization is necessary to reduce time variations among drifting local clocks. The delays and, more importantly, delay variations in software and network medium access times contribute to the time fluctuations that challenge the accuracy of a variety of applications requiring timestamps and synchronization.

Time is quintessential in a variety of software applications. Time is used to determine *date and time-of-day*, *time interval*, and *data collection frequency*. *Date and time-of-day* describes a point in time; it is fundamental for data time-stamping and for logging information. Timestamps provide a scalable, efficient, and technically sound mechanism for merging information retrieved from the rising confluence of heterogeneous data streams. Additionally, securing applications and systems require the ability to log events at the exact moment they occur. *Time interval* is applicable when determining the duration of an event or action. In certain manufacturing process applications, time intervals must be maintained to ensure that the final product meets the required specifications. *Data collection frequency* indicates how often an event should occur within a window of time. For example, Hz describes the frequency of occurrence over a time interval of 1 second. In data collection applications, consistent sampling rates may need to be ensured to avoid missing pertinent measurements.

These time synchronization guidelines reflect the anticipated needs of the semiconductor industry. They were developed and reviewed by the SEMI Data Quality Task Force and reflect their collective experience and perspective. The working group includes representatives from device makers, equipment suppliers, software suppliers, universities, and research consortia. The working group is led by ISMI.

The main objective of the guidelines is to provide guidance to establish a factory time synchronization architecture for realizing effective, consistent data collection and time-stamping for future data analysis. To achieve this, these guidelines provide information to ensure proper synchronization architecture and key synchronization factors to achieve reliable, accurate synchronization. These guidelines provide recommendations for device makers and equipment/software suppliers to implement the proposed new SEMI time synchronization standard requirements. Guidance is also provided for effective data time-stamping while ensuring data quality. Lastly, these guidelines provide accuracy and precision requirements for e-Manufacturing applications.

### 3 REFERENCES

#### 3.1 Standard References

SEMI E5 – *SEMI Communications Standard 2 SECS-II Message Content*

SEMI E30 – *Generic Model for Communications and Control of Manufacturing Equipment*

SEMI E120 – *Specification for the Common Equipment Model (CEM)*

SEMI E125 – *Specification for Equipment Self Description (EqSD)*

SEMI E132 – *Specification for Equipment Client Authentication and Authorization*

SEMI E134 – *Specification for Data Collection Management (DCM)*

SEMI E138 – *XML Semiconductor Common Components*

SEMI E148 – *Specification for Time Synchronization and Definition of the TS-Clock Object*

#### 3.2 Other References

- [1] NTP Software Download, [UNIX-based platforms](#); [Windows platforms](#).
- [2] J. Burbank, J. Martin, and D. Mills, [The Network Time Protocol Version 4 Protocol Specification](#) DRAFT, IETF, February 2008.
- [3] [Precision Clock Synchronization Standard for Sensor/Actuator Networks](#), IEEE 1588 2<sup>nd</sup> Conference, IEC 61588 ballot, 2004.
- [4] D. Mills, [Monitoring Options](#), University of Delaware, July 2005.
- [5] D. Mills, [Simple Network Time Protocol \(SNTP\) Version 4 for IPv4, IPv6, and OSI](#), IETF, RFC 4330, January 2006.
- [6] [Numeric Representation of Dates and Times](#), ISO 8601:2004, January 2003.
- [7] G. Crispieri, H. Wohlwend, and Y.S. Li, [Using Network Time Protocol \(NTP\): Introduction and Recommended Practices](#), ISMI, Technology Transfer #06024736A-ENG, February 2006.

- [8] M. Wolf and C. Wicksteed, [Internet Standard For Date/Time Format](#), W3C, September 1997.
- [9] D. Mills, [Network Time Protocol \(NTP\) \(Version 3\) Specification, Implementation, and Analysis](#), IETF, RFC 1305, March 1992.
- [10] [The QNX Neutrino Microkernel](#), QNX Software Systems.
- [11] D. Mills, *Computer Network Time Synchronization: The Network Time Protocol*, CRC Press, March 2006.
- [12] [Guidelines For Providing Multimedia Timer Support](#), Windows Hardware Developer Central, September 20, 2002.
- [13] J. Moyne, J. Parrott, N. Kalappa, and Y.S. Li, “[Practical Aspects Impacting Time Synchronization Data Quality in Semiconductor Manufacturing](#),” *AEC/APC Symposium XVIII*, October 2006.
- [14] [IGS FAQ](#), NASA Jet Propulsion Laboratory, February 16, 2006.
- [15] Su, W. “Overview of Time Synchronization in Sensor Networks.” *Embedded Systems Handbook*, CRC Press, 2006, p.35–1 to 35–10.

#### 4 TERMINOLOGY

Absolute Accuracy	A value determined by the specified offset between a clock’s current time and the current time of Universal Time Coordinated (UTC) where UTC is the internationally recognized standard for ideal time
Accuracy	Closeness of the agreement between the result of a measurement and a true value of the measurand. Since the true value cannot be determined exactly, the measured or calculated value of highest available accuracy is typically taken to be the true value
ACTS	Automated Computer Time Service
ANSI	American National Standards Institute
Date	A unique value captured as a timescale progresses
DNS	Domain Name Service
Epoch	A static date of interest, such as the date an era originates
Era	A time span with a designated beginning
GLONASS	Russian Global Navigation Satellite System [14]
GNSS	Global Navigation Satellite Systems [14]
GPS	Global Positioning System
IEC	International Electrotechnical Commission
IERS	International Earth Rotation Service
IETF	Internet Engineering Task Force
ITS	Internet Time Service
LAN	Local Area Network

MCXO	Microcomputer-Compensated Crystal Oscillator
Minimum Data Sampling Interval	The frequency that a parameter value (or buffer of values) is sampled by an application as indicated in <i>ms</i> ; the value does not indicate the total bandwidth of the application
NIST	National Institute of Standards and Technology
NTP	Network Time Protocol
OCXO	Oven-Controlled Crystal Oscillator
PCI	Peripheral Component Interconnect
PKI	Public Key Infrastructure
Precision	The quality of repeatability of measurement data, customarily expressed in terms of the standard deviation of the extended set of measurement results from a well defined (adequately specified) measurement process in a state of statistical control
PTP	Precision Time Protocol – PTP is based on the IEEE 1588 (IEC 61588) standard
Relative Accuracy	A value determined by the clock’s nominal offset to absolute time within the context of the designated local factory time
Resolution	The smallest possible increment of time in the clock model (e.g., number of ticks per second)
SI Units	Système International d’Unités
SNTP	Simple Network Time Protocol
Stratum	A collection of time server layers that are used to time synchronize
Synchronization	The act of maintaining and adjusting clocks within a system such that all clocks within the system will have the same epoch and measurements of any time interval by any of the clocks will differ by no more than a specified uncertainty
TAI	International Atomic Time
TCXO	Temperature-Compensated Crystal Oscillator
Timescale	A continuum of values used to denote time with respect to a frame of reference
Timestamp	A compact representation for representing date and time. In Network Time Protocol (NTP), timestamps are associated with an era to provide unambiguous mapping to dates
UDP/IP	User Datagram Protocol/Internet Protocol
UTC	Universal Time Coordinated
UUID	Universal Unique Identifier
WAN	Wide-Area Network

## 5 KEY SEMICONDUCTOR INDUSTRY REQUIREMENTS AND RECOMMENDATIONS

The key requirements and recommendations for semiconductor manufacturing environments are summarized below.

### 5.1 Clock Synchronization

#### 5.1.1 Requirements

- All equipment clocks must be capable of displaying Universal Time Coordinated (UTC) time using ISO-8601 format as defined by the SEMI E148 (Section 6.1).
- The equipment clock must be accessible to a mainstream time distribution protocol. Network Time Protocol (NTP) is the recommended protocol for near-term factory clock synchronization deployment. Compliance with SEMI E148 requires use of NTP version 3 (Section 6.6).
- Equipment clock synchronization must be configurable by an administrative user (Section 6.7).
- Equipment must have accessible interfaces to enable or disable synchronization for individual pieces of equipment (Section 6.13).
- The equipment must have a readily accessible interface to determine clock and clock synchronization quality as defined in SEMI E148 (Section 6.13).
- The equipment must generate appropriate status messages as defined in SEMI E148 when the equipment clock deviates from the factory clock synchronization requirements (Section 6.13).
- The TS-Clock object interface values must be updated at each synchronization cycle (Section 6.13).
- Equipment must be responsible for synchronizing the time of all of its subsystem and subassembly clocks (Section 6.15).

#### 5.1.2 Recommendations

- The time synchronization system should be fault-tolerant. Each stratum level should have at a minimum a cluster of three servers with diverse time sources (Section 6.4).
- Diverse time sources are necessary to provide fault tolerance when one source is unreachable or fails. For example, stratum 1 servers should synchronize to more than one type of time source to enhance the robustness of the synchronization system (Section 6.5).
- Clients should operate in many-cast mode or be assigned several servers in case a single server is unreachable (Section 6.8).
- To minimize degradation of the time synchronization accuracy, the NTP hierarchy should include no more than five strata (Section 6.8.1).
- The synchronization frequency can be reduced by improving the quality of the internal clock. For applications with stringent timing requirements, internal clocks used to maintain

accurate time and date with respect to the provided factory server must be carefully selected (Section 6.9).

- NTP should dynamically adjust the synchronization interval when possible. Maintaining nominal 1 ms synchronization accuracy typically requires PC-quality clocks to be synchronized *at least* once every 15 minutes for systems with sufficient resolution that reside in the same low jitter subnet. The *minpoll* and *maxpoll* variables for each server should be adjusted to achieve the desired the accuracy (Section 6.11).
- For clocks that have not been synchronized recently, such as a system that has just been powered up, the time synchronization protocol should run for at least 5 minutes before running applications that require synchronized time. When the system starts up, the synchronization frequency is typically once every 1 minute until the clock stabilizes (Section 6.11).
- For optimal clock synchronization performance, network bandwidth should be sufficient and router hops should be minimized to maintain low network jitter (Section 6.12).
- The clock synchronization performance should be periodically monitored to ensure equipment or factory application clocks satisfy the factory synchronization requirements (Section 6.13).
- Factory network security measures for clock synchronization deployment should be followed. Firewall rules must be set when accessing external servers. NTP uses User Datagram Protocol port 123. NTP also provides authentication and access control mechanisms for authenticating server time messages. Authentication can be configured using the key or autokey command for each peer, server, broadcast, and *manycastclient* assignment in the configuration file (Section 6.14).
- The resolution of the equipment clock should at least reflect the most stringent accuracy requirements, if not better. For example, the clock resolution for arc detection applications should be on the order of 100  $\mu$ s, since the application requires timing intervals of 500  $\mu$ s (Appendix A – Table A-2).

## 5.2 Time-Stamping

### 5.2.1 Requirements

- Reported timestamps, including the DateTime attribute in the TS-Clock object, must represent the actual accuracy resolution of the internal clock (Section 6.16).
- The software time-stamping format must adhere to SEMI E148 (Section 6.16.2).
- The resolution of the timestamp should reflect the accuracy that can be achieved by the clock generating the timestamp. If the accuracy of the clock is less than the resolution of the timestamp format, the remaining places available should be 0 (Section 6.16.4).

### 5.2.2 Recommendations

- The epoch of digital equipment clocks must converge to a single time base, preferably the NTP time base. The NTP time base corresponds to 0h 1 January 1900 (Section 6.16.1).
- Data generated from the equipment should be time-stamped as close to the point of data measurement as possible (Section 6.16.3).

- Because of significant latency variability between data measurement and data time-stamping, it is beneficial to compute the latencies when necessary and possible (Section 6.16.3).

## 6 TIME SYNCHRONIZATION GUIDELINES

### 6.1 Timescale

*Factory and manufacturing equipment must use the UTC timescale for time synchronization.*

**Implementer:** Equipment supplier, third-party software suppliers, and device makers

**User:** Equipment suppliers, third-party software suppliers, and device makers

**Standards:** SEMI E148

**Remarks:**

- To extrapolate the local time, one has to add or subtract a constant based on the local time zone.
- In areas where Daylight Savings Time is observed, the factory's local time moves ahead 1 hour when Daylight Savings Time begins. The UTC time remains the same, but the local time adjustment will include an additional hour. For example, if U.S. Eastern Standard Time is -5 hours from UTC, the U.S. Eastern Daylight Savings time will be -4 hours from UTC.
- To follow the UTC timescale, clocks must be able to receive and handle leap second information. Leap seconds are inserted on either June 30 or December 31, if an announcement of a leap second is made by the IERS. A positive leap second will appear as 23:59:60 before 00:00:00 of the next day. A negative leap second will appear as 23:59:58 before 00:00:00 of the next day.
- Global positioning system (GPS) and International Atomic Time (TAI) timescales do not have leaps seconds, but can be readily converted to UTC timescale. Leap second announcements can be received from GPS receivers, external radio clocks, and upstream NTP servers.
- Clock slewing is the preferred method of handling leap seconds in operating systems. Users should be aware that different operating systems handle leap seconds differently. Some allow clocks to step back 1 second during a leap second; however, stepping back may perturb systems relying on precise ordering of events. UNIX- and Linux-based system kernels handle leap seconds by slewing the system clock monotonically. Windows systems by default do not recognize leap seconds; however, an NTP client can be installed that can receive leap second announcements and use slewing to adjust the system time.
- The display format should adhere to ISO 8601 specifications as defined in SEMI E148.

**Background:** UTC is *universal* time; i.e., it has the same value anywhere in the world. The time value for UTC is determined based on a *coordination* of input from more than 50 national standards institutions around the world by the Bureau International des Poids et Mesures (BIPM)

in Sevres, France, to create TAI. The TAI is translated to UTC based on advice from the International Earth Rotation Service (IERS) in Paris, France, which serves as the authority for dictating leap seconds. TAI and UTC differ by an integral number of seconds.

## 6.2 Time Servers

*The factory is responsible for providing local time servers that disseminate UTC time to all computers, applications, and equipment requiring synchronized clocks.*

**Implementer:** Device makers

**User:** Device makers, equipment and third-party software suppliers

**Standards:** SEMI E148

**Remarks:**

- Factory time servers act as a time source for other factory distributed systems.
- It is recommended to have a cluster of three or more time server peers attached to accurate time source(s) for synchronizing all devices within the factory. Diversifying the time sources when a single time source is unreachable is also beneficial.
- Factory time servers must generally be considered more reliable, precise, and accurate than the clocks they are synchronizing. The resolution and accuracy of the time server clock will limit the accuracy achievable in the system of distributed clocks.
- Having a resident primary time server provides a more secure and consistent source for all the local devices, while reducing Internet bandwidth usage and network firewall management complexity for client nodes.
- Depending on the protocol used, the servers can be determined statically (NTP) or dynamically (PTP). The root time server, stratum 1 or 2 servers and grandmaster clocks for NTP and PTP, respectively, would be the definitive source of time for the factory distributed environment.
- Servers typically receive their timing information from a highly reliable source such as a GPS or radio signals traceable to a national or international metrology institute.
- Factories can configure a router's (or switch's) time to be an authoritative time server for each device on its subnet. System clocks in routers maintain time internally based on UTC, but can be configured to display the time in the local time zone. Routers can distribute time to nodes on their network path. Some routers can also be attached to a radio or atomic clock.

**Background:** Dedicated time servers are commercially available for NTP, while PTP-compliant products are beginning to emerge. Factories can choose to create their own NTP time servers by selecting a traceable time source and installing an NTP server software *daemon*, which is freely available for several platforms including UNIX and UNIX-based systems [1]. NTP is offered as a *service* for Windows systems in default installations. Reference NTP implementation ports for Windows are also freely available [1].

Commercially available time servers with integrated GPS or radio receivers are available. Examples<sup>1</sup> of such suppliers include

- [Meinberg](#)
- [Symmetricom](#)
- [Spectracom](#)
- [EndRun](#)

Another example option that results in an effective NTP primary server is simply a FreeBSD surplus Pentium with a GPS navigation receiver such as Garmin [11].

### 6.3 Traceability

*The reference UTC source should be traceable to either an international or national time reference standard.*

**Implementer:** Device makers

**User:** Device makers, equipment, and third-party software suppliers

**Standards:** SEMI E148

**Remarks:**

- A measurement is *traceable* if it is made based on a global or national standard reference and the uncertainty of the measurement is known and documented.

**Background:** Time is one of the seven base units of Système International d'Unités (SI) measurements; therefore, many national and international metrology institutions provide the definition of a second and disseminate a reference timescale.

- *International:* Bureau International des Poids de Mesures (BIPM) collects time information from over 200 atomic clocks in 50 national standards bodies to provide a single global time standard.
- *National:* For example, the National Institute of Standards and Technology (NIST) maintains an ensemble of cesium oscillators to provide a UTC (NIST) source that can serve as both a time and frequency reference. The UTC (NIST) reference is adjusted occasionally to maintain agreement with the international timescale, but never more than 2 ns per day.

---

<sup>1</sup> None of the suppliers identified in this report are recommended or endorsed, but are offered only as examples.

## 6.4 Fault Tolerance

*The time synchronization time servers in the factory should be fault-tolerant.*

**Implementer:** Device makers

**User:** Device makers, equipment, and third-party software suppliers

**Standards:** SEMI E148

**Remarks:** The key is to ensure the time synchronization network is fault-tolerant when one source fails.

- For servers providing time synchronization information to time-critical applications, the factory should provide a cluster comprising three or more servers in case a single server goes off line.
- The time sources and network paths to the time source must be diversified to achieve sufficient fault tolerance. If GPS signals are momentarily lost, it is useful to have redundant time servers synchronized to one another or to have additional tiers of time servers based on their achievable accuracy for determining the actual current time.
- NTP clients should synchronize to several servers instead of relying on a single server.
- Time servers can also have integrated time references, such as a high quality atomic oscillator or a radio clock.

## 6.5 Time Source

*The factory is responsible for selecting a traceable reference time source.*

**Implementer:** Device makers

**User:** Device makers, equipment, and third-party software suppliers

**Standards:** SEMI E148

**Remarks:**

- Distributed time synchronization protocols typically require an accurate UTC time source for the local time server. Having diverse and redundant sources to ensure a sufficient level of fault-tolerance is recommended.
- *Reference time source*, the source of time to which other network nodes synchronize, must be highly accurate and precise. The quality of the synchronization is largely affected by the quality of the clock source.
- Equipping every node with an expensive quality clock is often too costly to be justifiable. A more cost-effective solution involves setting up servers attached to a reference clock that disseminates the time to other nodes with clocks of lesser stability and accuracy. Time services provided by a national standards institute, such as NIST, have the additional benefit of traceability to a national standard frequency and timescale. NIST provides a UTC time reference for GPS, radio, Internet, and dial-up time sources.

**Background:** Reference time sources typically include GPS or radio receivers. Accurate time sources can be acquired in various ways (see below).

### 6.5.1 Global Navigation Satellite System Receivers

Global Navigation Satellite Systems (GNSSs) [14], such as the United States' GPS and Russia's GLObal NAVigation Satellite System (GLONASS), using a quality receiver can provide about 1  $\mu$ s UTC accuracy from an atomic clock to the local time server receiving the signal. GPS receivers also provide a secure and reliable communication channel, but when service is interrupted it is safer to rely on a backup time source. GPS-based clocks currently depend on U.S. operation of the GPS satellite constellation and require line-of-sight. The Galileo system, a second GNSS source, is currently being devised by the European Union with a planned constellation of 30 satellites providing improved precise timing source and location services. Galileo is expected to be fully operational by the end of 2010.

Typical GPS receivers are able to track 8–12 satellites simultaneously and provide time and frequency signals based on an average of the data obtained from all the satellites in view. The receivers provide the signal by RS-232 or other standard interface. GPS time and frequency sources are available from commercial suppliers including Symmetricom, Meinberg, and Spectracom. Commercial GPS receivers are available as integrated components in a time server, Peripheral Component Interconnect (PCI) cards, or as an external attachment.

### 6.5.2 Radio Receivers

Reception of radio broadcasts is limited by distance as well as uncertainties and variability in radio signal propagation. Radio receivers acquire periodic time signals from broadcasting stations. NIST has two station locations, WWV and WWVB in Fort Collins, Colorado, and WWVH in Kauai, Hawaii. WWV and WWVH require high frequency (HF) receivers while WWVB requires a low frequency (LF) receiver. WWVB covers the entire continental U.S. and parts of Canada, while WWV and WWVH together provide almost worldwide coverage. In Europe, DCF77 and Rugby also broadcast time signals from near Frankfurt, Germany, and the U.K. respectively. Japan radio time is acquired from JJY and is similar to WWVB.

Absolute accuracies range from 1–20 ms for HF receivers and 0.1–15 ms for LF receivers. Another key advantage of LF signals, such as WWVB, is that the signals can be received through simple indoor antennas; however, to consistently acquire the signal requires an outdoor antenna for optimal results. To obtain 0.1 ms of UTC, the receivers need to be able to estimate the path delay by determining the distance between the receiving site and WWVB and the time required for the radio signal to travel that distance. The receivers should then be able to compensate for that delay.

### 6.5.3 Internet

Using NTP or the NIST Internet Time Service (ITS), a factory time server can acquire the UTC time source through the Internet. The primary server for NIST that propagates the atomic time is *nist.time.gov*. Because of the great variability of Internet traffic, it is recommended not to rely solely on Internet-based time.

Any transmission through the Internet inherently has delays on the order of 50 ms to seconds because of the uncertainty of routing protocols. The accuracies achieved range from tens of ms to seconds. The Internet also requires an additional UDP/IP port to be opened for receiving the

signal, which becomes an additional vulnerability if the rest of the company is protected by a firewall.

#### 6.5.4 Dial-Up

A dedicated dial-up line can provide an accurate time source within 15 ms absolute accuracy. NIST provides the automated computer time service (ACTS) with a dial-up number. Since time packets do not arrive in large files, the modem connection is often sufficient for providing a source of traceable UTC time. The local analog modem can also send a corresponding message to ACTS to allow the service to estimate the delay of the telephone communication. A dial-up source is not recommended for time-critical applications requiring less than 15 ms accuracy.

### 6.6 Synchronization Protocol

*The factory is responsible for selecting a time synchronization protocol to propagate traceable UTC time throughout factory systems. Equipment suppliers are responsible for supporting at least one of the mainstream time synchronization protocols. NTP is currently the preferred protocol.*

**Implementer:** Device makers, equipment suppliers

**User:** Device makers

**Standards:** SEMI E148

**Remarks:**

- The mainstream distributed time synchronization protocols are NTP and PTP. Fully exploiting the advantages of PTP requires IEEE 1588-compliant hardware. As NTP sufficiently meets current requirements without additional hardware components, NTP is the recommended protocol for the near-term.
- It is recommended to use the latest stable implementation of the respective protocol for optimal accuracy and security. The latest stable NTP reference implementations are available at [ntp.isc.org](http://ntp.isc.org).
- The time protocol selected should carefully consider applications requirements and cost constraints.
- NTP and PTP can co-exist in a factory time synchronization architecture.

#### 6.6.1 NTP

NTP is a time-tested and ubiquitous protocol for synchronizing distributed systems through the Internet. NTP supports virtually all platforms either as part of the operating system (UNIX, MacOSX) or as a piece of client software (Windows). NTP version 4, which is generally backwards-compatible with the previous versions except for version 1, is currently the latest reference implementation.

Detailed NTP specifications are published as Request for Comments (RFCs) under the Internet Engineering Task Force (IETF). RFC 1305 [9] provides the NTP version 3 specification, and RFC 4330 provides the SNTP version 4 specification. The current working draft of NTP version 4 and reference implementations are available, but the RFC is currently a work in progress, and if approved, RFC 4330 and RFC 1305 will become obsolete [2].

On a wide-area network (WAN), NTP provides time synchronization from 10–50 ms, nominally, while 1 ms absolute accuracy can be achieved if a GPS is used as the time source within a local area network (LAN). NTP has been designed to survive network, node, software, and clock errors and failures. Security mechanisms for client/server authentication and authorization have also been specified to prevent the network or system from being compromised.

### 6.6.2 PTP

IEEE 1588 provides the specification for the PTP, which has been designed for industrial sensor/actuator LANs characterized by small subnets for optimal performance. The key drivers to developing a new synchronization protocol were primarily the accuracy and synchronization convergence demands of high performance, cutting-edge automation equipment, and industrial networks.

Current semiconductor industry requirements do not require the accuracies achievable by PTP; however, as solutions become available and time synchronization requirements are pushed to the limits of NTP, it can be useful to have other solutions available. IEEE 1588 is a hybrid solution comprising both hardware and software elements that can be implemented on any packet-based protocol capable of multicasting. The hardware elements enhance the synchronization performance by minimizing the variable latencies occurring in non-deterministic elements of the network. Including both the hardware and software elements of IEEE 1588 can provide sub-millisecond synchronization accuracy among local clocks using the same network that transports process data. IEEE 1588 has been gaining momentum in both the U.S. and abroad. Both ANSI and IEC have adopted the standard. The standard is also called IEC 61588 [3]. Commercially available IEEE 1588 products include time servers, Ethernet switches, PCI-based network interface cards, microprocessors, etc., by a variety of suppliers including Meinberg, Symmetricom, Hirschmann, National Instruments, and Intel. Table 1 compares the basic characteristics of NTP and PTP.

Version 2 is currently an approved draft, but is still awaiting publication as an IEEE standard. IEEE 1588 version 2 provides improvements in time synchronization accuracy by adding the concept of *transparent clocks*. Transparent clocks are devices that measure the delay of the PTP message being sent while it is being transmitted by the device. Version 2 allocates a *correction field* in the PTP message header to record the delay measurement data, therefore allowing any delay incurred to be calculated while synchronizing the slave clock.

In addition to improvements in accuracy, another major change is the specification of security for IEEE 1588. While security is included in the extension, it is not a required part of IEEE 1588, and its impact on performance is still being studied. PTP message authentication is verified by an integrity check value to determine if the time can be trusted. To enable security, the secure flag in the header must be enabled.

**Table 1 NTP/IEEE 1588 Comparison**

	<b>NTP</b>	<b>IEEE 1588</b>
Accuracy in LAN	milliseconds sub-milliseconds with nanokernel	sub-milliseconds with software sub-microsecond with hardware
Distribution Model	Hierarchical Client-Server/Peer Ensemble Multicast Manycast	Master-Slave Multicast
Network Topology	WAN LAN	LAN with small subnets WAN application under study
Security	Autonomous authentication/authorization	Message and source authentication included as an experimental security extension in IEEE 1588 version 2
Implementation	Pure software Over 20 years Available on: PCs to Crays UNIX, Windows, VMS, Linux	Hardware/Software New standard Commercial products emerging

### 6.6.3 Hybrid (NTP and PTP)

Two protocols, such as PTP/IEEE 1588 and NTP/SNTP, have been designed to readily co-exist. Merely connecting the reference time source to an IEEE 1588 boundary clock and to the respective NTP servers synchronizes the systems. NTP and SNTP time information is transparent to each other. Figure 1 illustrates how the protocols can co-exist.

Between IEEE 1588-enabled systems and NTP systems, a *gateway* that translates the packet information from one protocol to the other is required. Figure 2 illustrates how a gateway can be used between a boundary clock and NTP time servers.

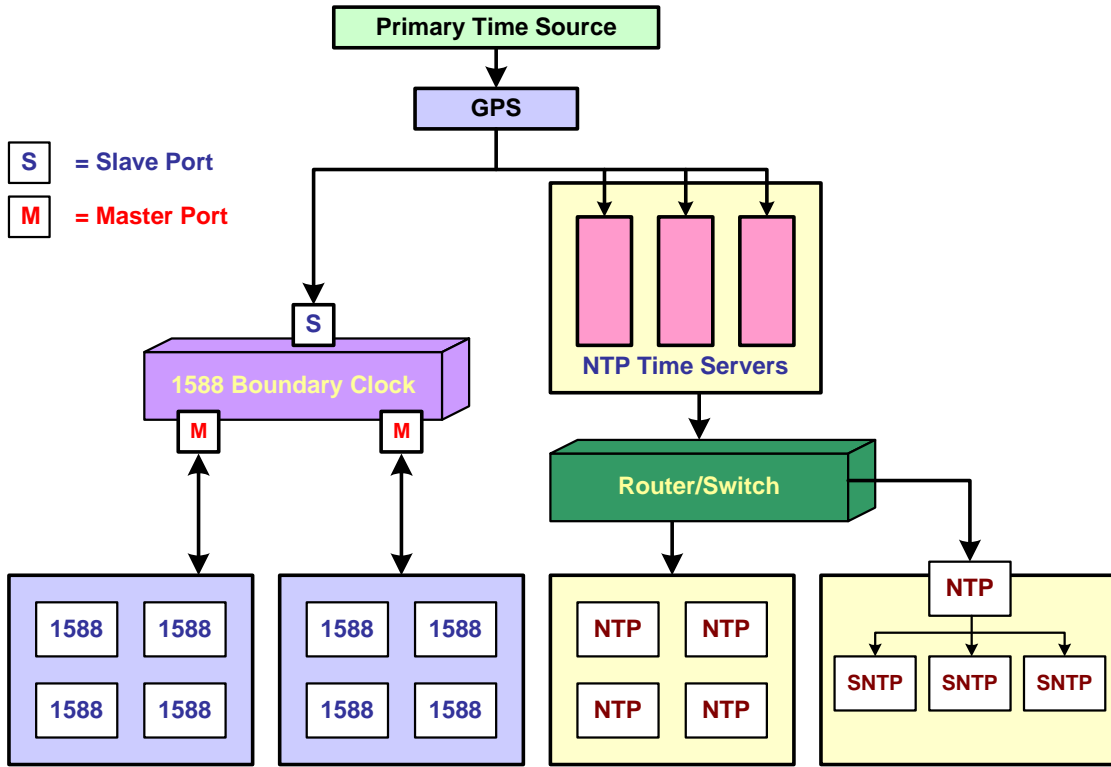


Figure 1 Co-Existing NTP/PTP Time Synchronization Protocols

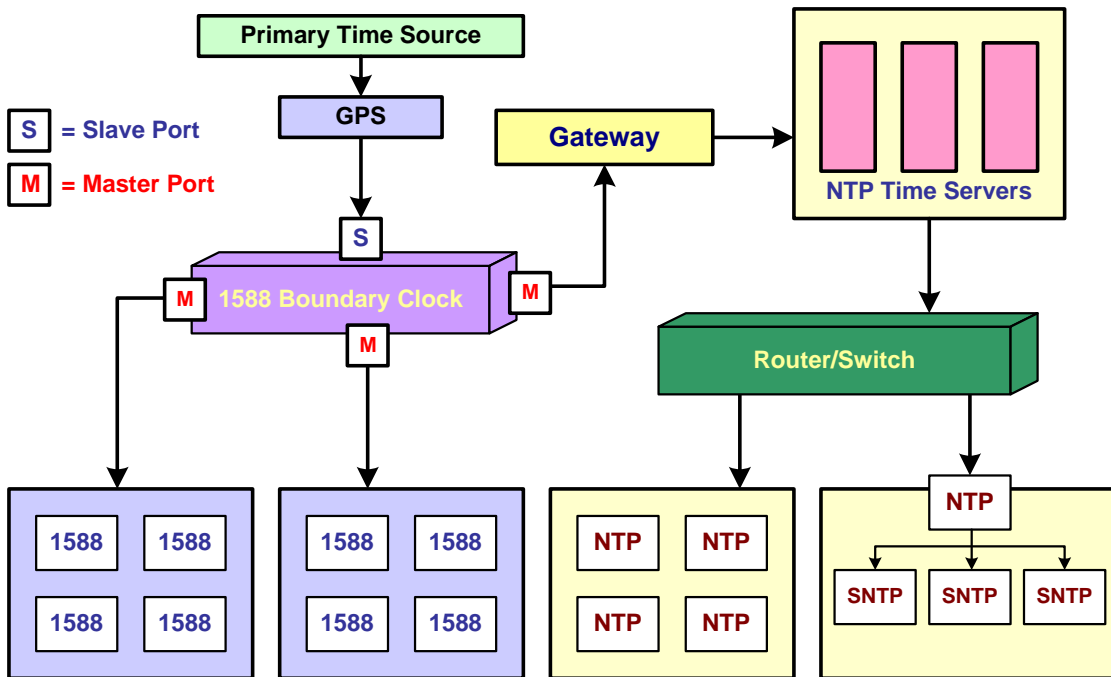


Figure 2 PTP-Based Time Synchronization With NTP Gateway

## 6.7 Client Configuration

*Time synchronization clients should allow users to configure its time synchronization software.*

**Implementer:** Equipment and third-party software suppliers, device makers

**User:** Device makers, equipment and third-party software suppliers

**Standards:** SEMI E148

**Remarks:**

- The client must support the mainstream time synchronization protocol (i.e., NTP). Users and implementers should remain aware of PTP development as time synchronization demands become more stringent in the future.
- The client must allow administrators to configure the servers with which the client will synchronize.
- Ideally, clients should synchronize with three different servers for the most robust clock synchronization quality. When a single server fails, two servers still remain to synchronize time. With a single server or two servers, no majority clique can exist to determine a failed clock.
- NTP also recommends the use of a Domain Name Service (DNS) to perform network load-balancing among server peers by changing IP address bindings to canonical names [5].

**Background:** NTP properties can be configured in an *ntp.conf* file, which typically resides in the /etc directory on UNIX-based systems or in C:\WINDOWS\system32\drivers\etc on Windows systems. Upon first use of *ntpd*, the file configuration must be created with an editor if it does not already exist. A graphical user interface (GUI) can also be developed to ease access to the file for configuration purposes. Some of the key configuration properties include the following:

- *Server:* IP address or hostname of the server assigned to the client. A server resides at a lower stratum level than the client.
- *Peer:* IP address or hostname of the peer assigned to the client. A peer resides on the same stratum level.
- *Broadcast:* IP address or hostname of the broadcast server. The association mode between the client and server will be broadcast mode where the server has a persistent broadcast/multicast address to propagate information to all the respective clients.
- *Manycastclient:* IP address or hostname of the manycast servers. Here the client broadcasts to a broadcast/multicast address and analyzes the responses to determine the best servers it will synchronize to. The result of this setting is client-server synchronization capability with the most accurate time server available to the client.

Each configuration command has associated options. Such options that should be available include the following:

- *Autokey*: Requires all NTP packets to include authentication information.
- *Burst*: A series of eight packets instead of one per synchronization (sent every 2 sec) when the server is reachable.
- *IBurst*: A series of eight packets per synchronization when the server is *not* reachable.
- *Minpoll*: The minimum frequency when using dynamic polling intervals. It is recommended to use the default *minpoll* value of 6, which corresponds to 64 sec.
- *Maxpoll*: The maximum polling frequency when using dynamic polling intervals. It is recommended to use the default *maxpoll* value of 10, which corresponds to 1024 sec.

## 6.8 Synchronization Architecture

*Achieving 1 ms time synchronization accuracy requires carefully architected time synchronization system.*

**Implementer:** Device makers

**User:** Device makers, equipment and third-party software suppliers.

**Standards:** SEMI E148

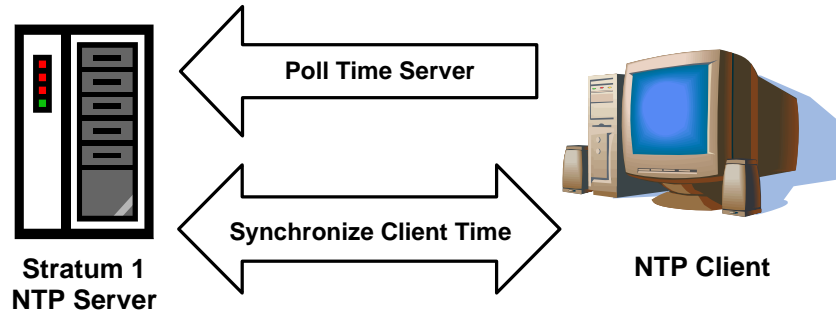
**Remarks:**

- Clients should be able to exchange the local time to the server whenever processing and network bandwidth are available. In terms of NTP, the client-server mode is preferred over broadcast or multicast modes. Configurations with clients that can only listen (e.g., broadcast mode) can degrade the synchronization system, especially in large networks, since the delays can vary significantly and the client will not be able to calculate propagation delay with respect to the server and itself.
- The architecture must support the organization based on the size and topology of the network. Other network factors to consider when designing the time synchronization architecture include support for redundancy, multiple subnets, large data volumes, and specialized time-sensitive applications.
- In broadcast and multicast mode, proper filters should be used such that only the intended servers or clients receive the messages.

**Background:** Association Modes

- ***Client/Server***

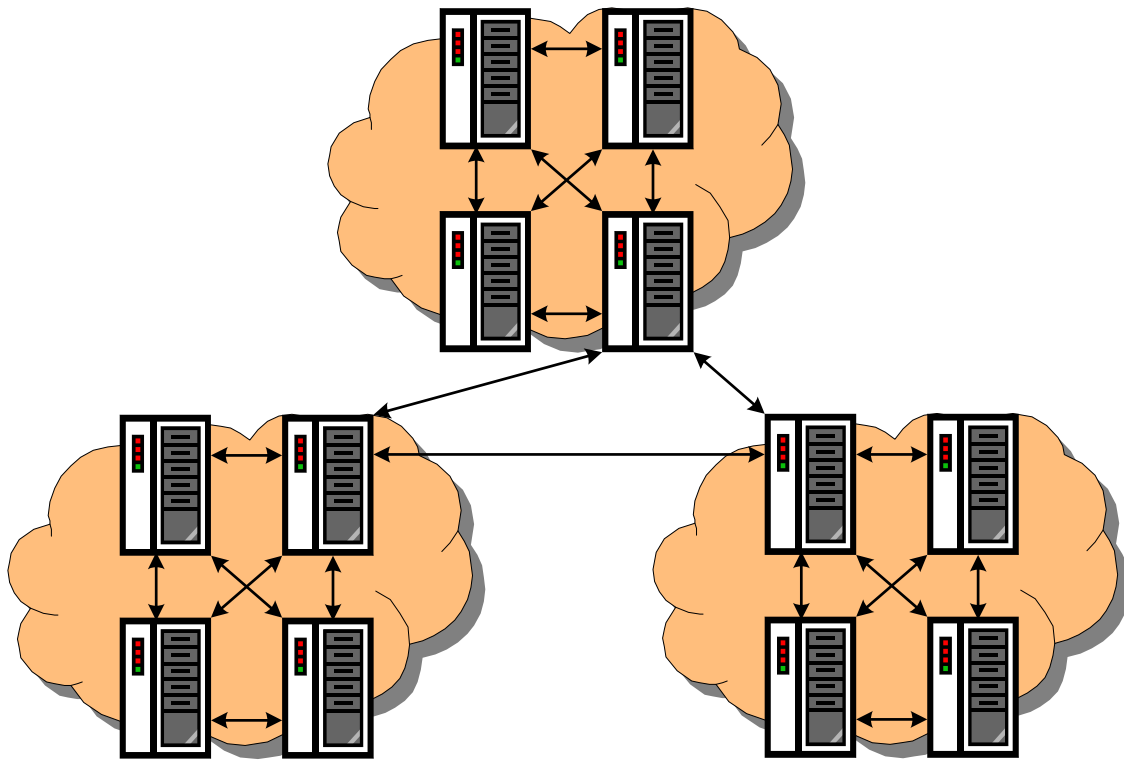
In the client/server mode, the client depends on a server within the group and no member in the same group and stratum can synchronize to the client. The client makes a request to the server by periodically polling the server for time information and expects a reply in the future (Figure 3). This set-up can generally withstand malfunction and protocol security attacks better than others.



**Figure 3 Client-Server Mode**

- *Symmetric Active/Passive*

Symmetric mode allows for a group of peers, typically in the lower strata, to operate as mutually redundant NTP servers (i.e., they back up each other). Each peer can be equipped with a primary reference source such as a GPS or be synchronized to another group of mutually redundant servers at the same stratum. In symmetric mode, a host polls other hosts on the same stratum, but also listens for and responds to other hosts on the same stratum. The goal of this mode is to provide a system of checks and balances among a group of servers sharing the same stratum to ensure it is not propagating poor timing information to higher stratum servers or clients. If one peer loses its time source or ceases to exist on the network, the remaining peers automatically reconfigure. In large NTP deployments, the recommended topology is to include clusters of lower stratum level servers (Figure 4).

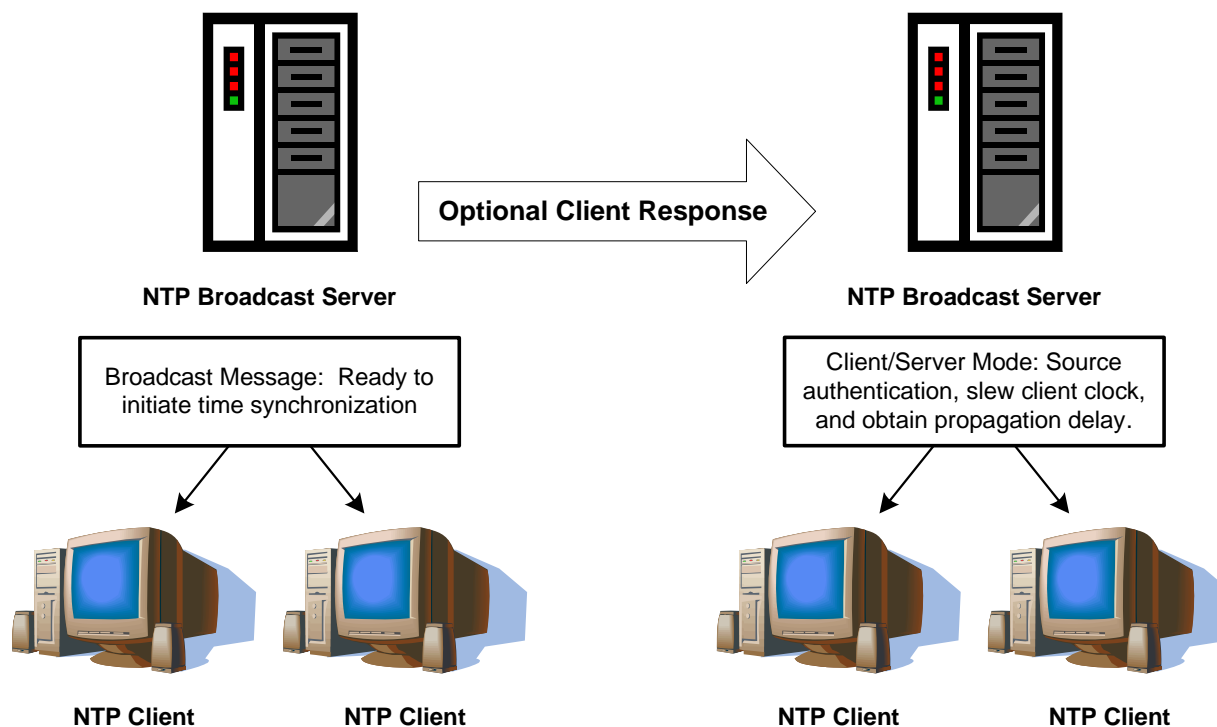


**Figure 4 Symmetric Active/Passive Mode**

- **Broadcast/Multicast Mode**

When accuracy and reliability of time information is not as stringent for a set of clients, this set of clients can be configured to rely on broadcast or multicast NTP messages. The trade-off lies in management complexity versus optimal NTP synchronization performance. The NTP topology is typically a set of NTP servers with a large group of clients all on the local subnet. Broadcast and multicast modes allow the administrator to use the same configuration file for many clients since clients are configured with the same broadcast server. Clients listening to broadcast and multicast messages are not required to respond to the server. However, configuration on the client side allows the client to set up an exchange with the server to improve estimates of the message propagation delay.

In broadcast mode, the server broadcasts a message to all of its clients to initiate synchronization (Figure 5). Following the receipt of the broadcast message, a client waits at random intervals before attempting to respond to the server to avoid a bombardment of messages if all the clients synchronized at once. The client and server continue to exchange a series of messages to authenticate the source, set the clock, and obtain an estimate of the roundtrip delay until the client time is set, at which time the exchange terminates; however, if the server does not respond, the client will use a default propagation delay to correct its clock.

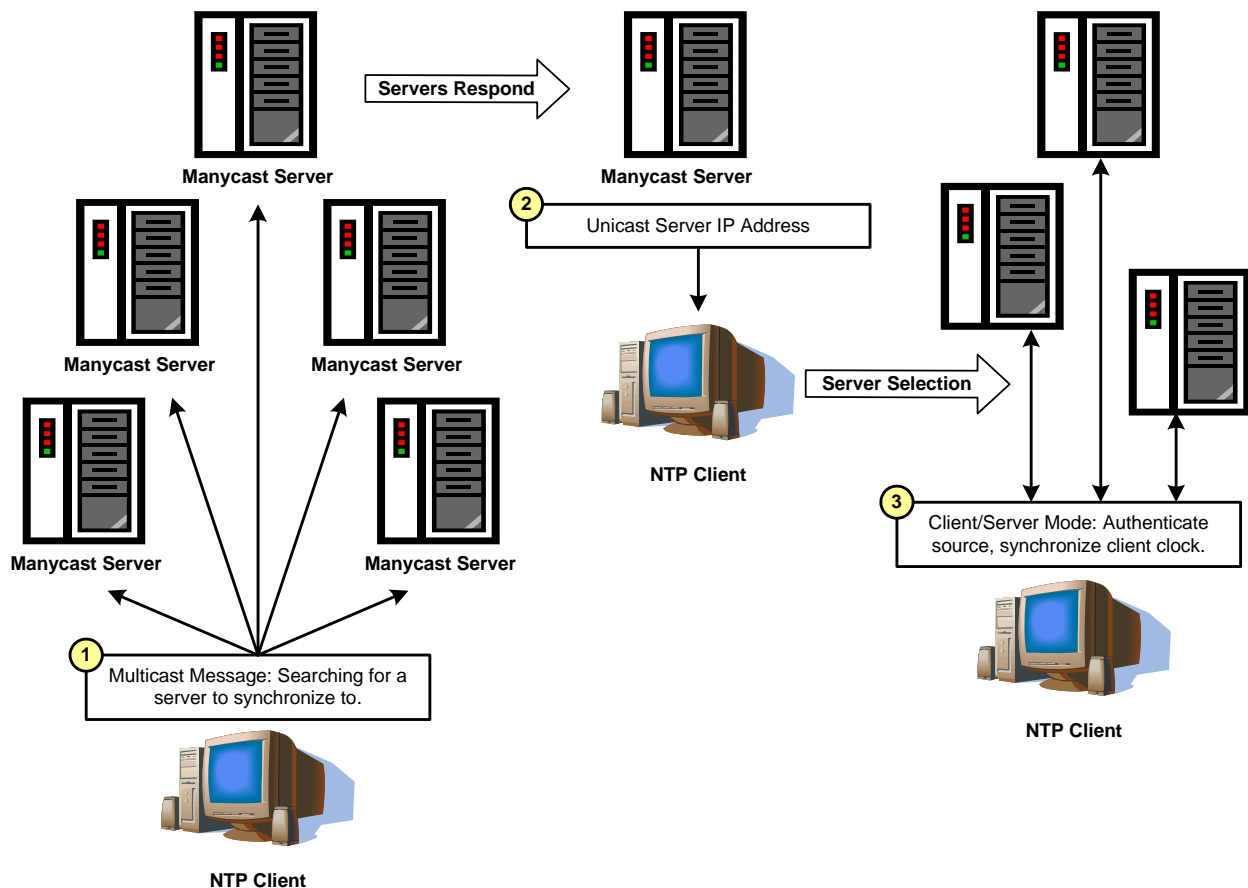


**Figure 5 Broadcast/Multicast Mode**

- **Manycast Mode**

As a new feature in NTP version 4, the manycast association mode is designed to enhance the reliability and ease of administration of the protocol. In manycast mode, NTP clients survey prospective servers. The clients can evaluate the time values of several servers using an expanding ring search of the IP multicast tree before deciding to synchronize with a set of the three most accurate and stable servers within its range as determined by the client-side NTP software.

As depicted in Figure 6, the client initiates a manycast exchange by multicasting messages at intervals with a specified time-to-live (TTL). A manycast server at an equal or lower stratum level receiving the message replies to the client with its address. Once the client receives the IP address, the client can establish a unicast exchange and operate in client/server mode. The goal is to enable the use of the best available server and to automatically reconfigure when a current server should fail. The manycast mode is especially useful in ad hoc networks, where mobile, wireless systems can synchronize to time servers within its range. However, wired systems can also benefit from the flexibility and robustness of the manycast paradigm.



**Figure 6 Manycast Mode**

### 6.8.1 Stratum Levels

*Maintaining a minimum number of stratum levels is recommended to preserve synchronization quality.*

**Implementer:** Device makers, equipment suppliers.

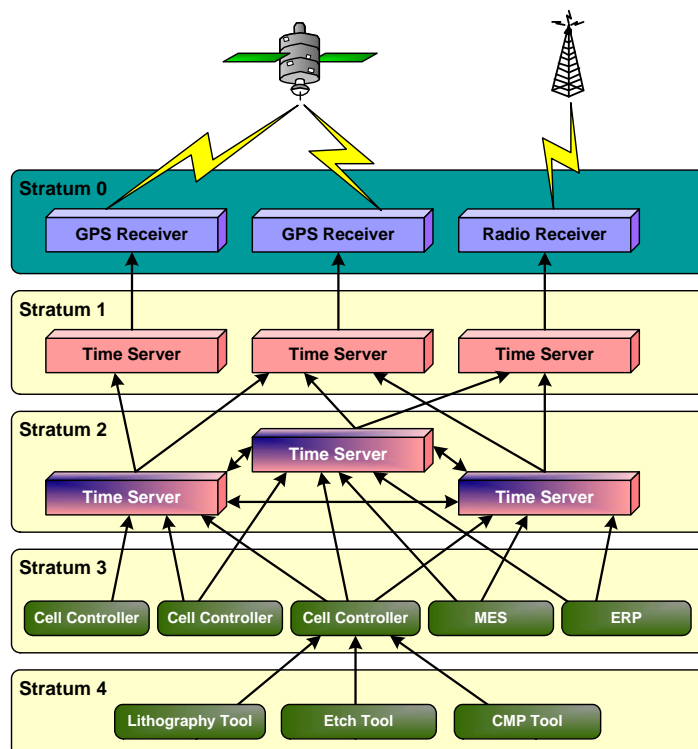
**User:** Device makers, equipment and third-party software suppliers.

**Standards:** SEMI E148

**Remarks:**

- In general, maintaining fewer stratum levels in a network will ensure the overall quality of the synchronization system, because synchronization quality degrades with each subsequent stratum level.
- For typical enterprise environments, approximately four stratum levels are recommended. In semiconductor manufacturing, the number of stratum levels should not exceed five. However, if the data flows (systems that need to merge data into a single application for analysis) are on the same stratum and do not require global accuracy, then the system would not necessarily benefit from residing at a lower stratum.

**Background:** The NTP *stratum* of a time server indicates how many levels a node is away from an authoritative time source. Figure 7 illustrates an example of how NTP strata can be implemented within a factory. For instance, a stratum 1 time server has direct access to a UTC



**Figure 7** Example NTP Hierarchy for Factory Distributed Time Synchronization

time source such as a radio or atomic clock. NTP strata range from 0–16, with stratum 0 being an atomic frequency source and stratum 1 as the time server synchronized to the atomic frequency source by a GPS or radio receiver.

In a large enterprise, the layered hierarchy ensures greater extensibility of growing IT networks. Stratum 2 servers can synchronize to a single or multiple stratum 1 servers and can also be organized to synchronize among other stratum 2 servers to verify all the servers are propagating the same time to the rest of the network. A stratum 3 NTP host can typically serve multiple lower level devices or the clients of a particular system residing at stratum 4. For example, an equipment host (cell controller) could be responsible for synchronizing multiple pieces of equipment (e.g., lithography, etch, chemical mechanical polishing, etc.). The equipment serves as a stratum 4 host to its subsystems at the stratum 5 level, such as the sensors, data client, integrated metrology, and other modules within the equipment. The factory must design the hierarchy based on its synchronization requirements and network.

## 6.9 Clock Quality

*A stable, high resolution clock improves time synchronization by reducing the time synchronization frequency needed and network bandwidth.*

**Implementer:** Equipment and third-party software suppliers

**User:** Device makers

**Standards:** SEMI E148

**Remarks:**

- NTP does not require specific hardware clock requirements; it is intended to mitigate the instability of typical computer clocks. However, quality clocks can reduce the network traffic significantly.
- If PTP is used, clock requirements include oscillator frequency accuracy and stability to within  $\pm 0.01\%$  of the SI second over its rated temperature range.
- For applications requiring 1 ms accuracy with minimal network and processor bandwidth available, using improved oscillators as specified for PTP is suggested.

**Background:** Quality clocks require less frequent synchronizations, allowing the factory to preserve available network and node resources and making the synchronization less susceptible to network disturbances. For example, less processor time needs to be dedicated to synchronizing the clock. Clocks are comprised of an oscillator and a counter. The low quality of the oscillator is the main reason why commodity clocks need to be synchronized. The inherent frequency can be susceptible to input voltage to the clock, inherent clock properties such as aging, and ambient factors such as temperature and humidity.

The quality of the clock can be characterized by its resolution and stability. Commercial oscillators typically include stability and accuracy over a temperature range in their specifications. The achievable accuracy is limited by the resolution of the clock and its ability to continuously follow the second based on the SI units. Stability can be gauged by various methods. Stability is measured based on the *Allan Deviation*, which determines the frequency fluctuations over a period of time. In layman's terms, stability can be noted as the time it takes for the clock to deviate by a set period of time, such as the highest deviation within 1 day.

Because retro-fitting existing equipment with new clocks can be costly, factories and equipment suppliers should carefully consider the trade-offs between timing requirements and cost before using specialized oscillators in place of its cheaper counterparts.

### 6.9.1 Precision Oscillators

For computers, peripheral component interconnect (PCI) boards are available that are dedicated to providing accurate time. Because PCI boards follow a standard computer bus specification for attaching peripheral components, they can be readily integrated into common electronic systems. The boards typically contain better quality clocks than the common clocks available. Specialized industrial oscillators are also available.

### 6.9.2 Quartz Oscillators

The cost-effectiveness of quartz crystals makes them among the most widely used resonator in oscillators. Quartz clocks are limited by aging, which affects long-term stability. Short-term stability is limited by environmental factors including temperature, humidity, pressure, and vibrations. The hardware clocks within PCs typically lose 5 to 15 seconds per day. Specialized quartz oscillators include temperature compensated (TCXO), microcomputer compensated (MCXO), and oven controlled (OCXO). TCXO clocks are generally cheaper, while compensating for frequency changes caused by the temperature. TCXO clocks are often only used if a high stability over a wide temperature range is not required. MCXO uses digital techniques to mitigate frequency drifts and generally performs better than TCXO. OCXO provides the best stability by maintaining the oscillator in a temperature-controlled enclosure.

**Table 2 Clock Quality**

Clock	Accuracy (sec)	Types	Cost	Stability*
Commercial Atomic	$10^{-10} - 10^{-15}$	Cesium Rubidium	\$8,000 – \$40,000	1 day to 1 month
Industrial Quartz	$2X 10^{-6} - 2X 10^{-8}$	TCXO MCXO OCXO	\$100 – 2000	1 sec to 10 days
Basic Quartz	$1 - 10^{-3}$	Consumer Products	< \$1	> 1 ms

\* Time for clock to be off 1  $\mu$ sec.

## 6.10 Clock Resolution

*The resolution of the clock should be at least one order of magnitude greater than the most stringent application requirement relying on the clock.*

**Implementer:** Equipment suppliers

**User:** Device makers

**Standards:** SEMI E148

**Remarks:**

- The recommended resolution for applications requiring at least 1 ms is at least 100  $\mu$ s.

**Background:** The resolution of the clock not only limits the maximum accuracy of the clock, but also determines how fine of an adjustment the clock can make and at what levels it can start detecting deviations from the server time.

## 6.11 Synchronization Frequency

*The synchronization protocol (i.e., NTP) should dynamically determine the synchronization frequency necessary to achieve a sufficient level of accuracy.*

**Implementer:** Device makers, equipment and third-party software suppliers

**User:** Device makers, equipment and third-party software suppliers

**Standards:** SEMI E148

**Remarks:**

- The synchronization protocol must be allowed to run continuously and provide updated time information at rates sufficient to compensate for a significant drift of the available oscillators. Synchronizing once when a system is booted up is typically not sufficient after a period of time. Ideally, the system must allow the synchronization protocol to dynamically determine the frequency of synchronization due to fluctuations in clock behavior.
- In NTP, the user is able to specify the time synchronization frequency by configuring the polling interval or the frequency with which the client will make requests to the server for the time. Optimal polling intervals typically range between 1 and 20 minutes. When a system begins running NTP (e.g., when it is first booted up), the system clock will require more frequent synchronizations.
- The *minpoll* and *maxpoll* configuration settings must be adjusted for each server to reach the desired level of synchronization accuracy. By default, *minpoll* is set to 6 ( $2^6$  or 64 seconds) and *maxpoll* is set to 10 ( $2^{10}$  or 1024 seconds, which is about 17 minutes).
- Unsynchronized clocks (e.g., equipment recently started) should run NTP for at least 5 minutes and up to 1 hour to allow the clock to stabilize and to be sufficiently synchronized.

- Tool performance should not be affected by clock synchronization. The synchronization frequency should not negatively impact server, network, node, or synchronization performance. To ensure server and protocol performance, clients must use exponential back-off when servers do not respond within a reasonable time [4].
- Quality clocks require less synchronization frequency and therefore alleviate time synchronization server demands, client resources, and network traffic. However, quality clocks have a more significant upfront cost.
- To avoid synchronization during critical equipment processing, synchronization can be forced before a process start. Sufficient time should be allowed for the system clock to stabilize if it has not been synchronized recently or if the clock has a significant offset from the server(s).

## 6.12 Network Jitter

*Network jitter should be minimized.*

**Implementer:** Device makers

**User:** Device makers

**Standards:** SEMI E148

**Background:** Minimizing network jitter (variability in network delay) improves synchronization quality. Network clock synchronization protocols rely on consistent network delays because algorithms assume a symmetric network delay when calculating the difference between the actual time and the timestamp sent by the server.

Network traffic can be reduced by increasing the bandwidth of the network. Network variability can also be reduced by decreasing the number of router, hubs, and switches between the synchronization client and the time server. Routers introduce significant delay variability whereas hubs and switches are better.

Reducing synchronization frequency can also result in lower network traffic; however, synchronization frequency should be sufficient to satisfy application requirements. For example, if an application requires a synchronization accuracy of only 1 second, then a synchronization frequency of once a day would suffice for a typical computer clock.

The use of network bandwidth should be monitored to ensure time synchronization packets do not experience a great variability in transport time between the client and server. In some systems, even 25% utilization can result in significant delay variability [13].

### 6.13 Synchronization Performance

*Ensuring timestamp quality requires monitoring time synchronization performance.*

**Implementer:** Device makers

**User:** Device makers, equipment suppliers

**Standards:** SEMI E148

**Remarks:**

- The NTP program, *ntpq*, can be used to retrieve the TS-Clock object attributes. The *ntpq* software comes in the reference implementation and is freely available in both Windows and UNIX-based (UNIX, Linux, Mac, BSD, etc.) platforms.
  - To obtain the *offset* from the time server, the following command can be used:  
`ntpq -p`
  - To obtain the *precision*, *lastsynchronizedtime*, and *status*, the following command can be used: `ntpq -c rl`
  - To obtain the current *datetime*, the following command can be used: `ntpdate`
- Management of the time synchronization network requires the ability to monitor and control the NTP server and clients. The equipment clock must implement a clock interface to provide synchronization performance attributes to the user. A method to notify equipment users of clock drift outside of a specified tolerance level is also recommended.
- Network time servers must maintain a reliable level of accuracy with a UTC source.
- The processing load of each server must be as balanced as possible.
- A central workstation is recommended to provide a centralized point of control, although management can be accessed through any web interface. The workstation must connect securely to the time source and to its clients.
- All the time servers and clients must be uniquely identified for the management layer software to monitor, configure, and troubleshoot the servers and clients. In IEEE 1588, each clock has the universal unique identifier (UUID) as its unique identifier. In NTP, clients and servers are identified by their IP address.
- The time management application must be able to monitor all the synchronized clients and note the accuracies at specific intervals in time in case the quality of a timestamp comes into question.
- The management server must work with the synchronization protocol to ensure automatic failover of servers and multiple fallbacks of time sources. The management software must also prevent tampering with the network time mechanism. Time synchronization protocols, such as NTP and IEEE 1588, typically have management capabilities encapsulated into the protocol, making it easier for network administrators to properly maintain a synchronized network.

- Alarms and/or logs ensure that the network nodes were accurately synchronized at a specific point in time. A dedicated log server can ensure integrity of the clock synchronization and time-stamping. The auditing server must be able to individually identify and monitor the nodes, record time, synchronization interval, accuracy offset, and time source on the individual systems; provide easy retrieval of such vital information such as when a clock was last adjusted; compile synchronization information into audit records; and issue immediate alerts when a monitored machine fails or swerves from desired tolerances.

### 6.13.1 Time Synchronization Monitoring Tools

NTP provides features to enable continuous, long-term monitoring of server and client clock performance by generating log files of various clock and synchronization statistics. The *ntpd* configuration command *statistics* enable six types of clock statistical information to be tracked: *clockstats*, *cryptostats*, *loopstats*, *peerstats*, *rawstats*, and *sysstats* [1]. Rigorous monitoring is typically done on stratum 1 and 2 servers; as the stratum becomes higher, less monitoring is necessary. The *clockstats* option allows the reference time source clock driver statistics to be recorded. A clock driver can record statistics of the hardware clock in addition to the default information, which includes the date, time, hardware clock address, and the last time code was received from the clock. The *cryptostats* command allows monitoring of the public key protocol; however, this option currently requires the OpenSSL cryptographic software library. The *loopstats* file contains statistics about the NTP filtering algorithms and the host's clock stability. The *peerstats* file provides clock statistics data about each of the peers the current server associates with. The *rawstats* file contains a log of all the messages received from peers and clock drivers with which the host is synchronized.

Accessing the *sysstats* file provides information on *ntpd* statistics such as the time since the system was last rebooted, the number of packets received, the number of packets denied, and the number of packets matching the current NTP version.

## 6.14 Security Practices

*Adhering to enterprise security practices and additional security precautions will prevent accidental or malicious interference with system clock synchronization.*

**Implementer:** Device makers

**User:** Device makers

**Standards:** SEMI E148

**Remarks:**

- Using the security features available in NTP is recommended. NTP has provisions for both access control and authentication. Authentication provides a mechanism for clients to verify that a time server is known and trusted and not a system that is accidentally or maliciously masquerading as a server. Authentication is ensured by an initial exchange of keys, which systems are not required to exchange in advance. NTP version 4 contains the *Autokey* feature for authenticating NTP packets. NTP version 3 provides only symmetric key authentication, which is generally believed to be less secure because the client and the server use the same key.

- Enterprise security policies for time must be enforced just like any other network application. Firewall rules must be applied appropriately.

**Background:** Unauthorized access by computers external or internal to the network can introduce inaccurate times and thus degrade the quality of the synchronization scheme.

### 6.14.1 Firewalls

NTP by default uses UDP port 123 to send and receive NTP packets. In locations where the enterprise firewall does not allow external UDP traffic into the factory network, the factory can use an external time receiver (e.g., GPS, radio) and only internally disseminate NTP information by UDP.

### 6.14.2 Access Control

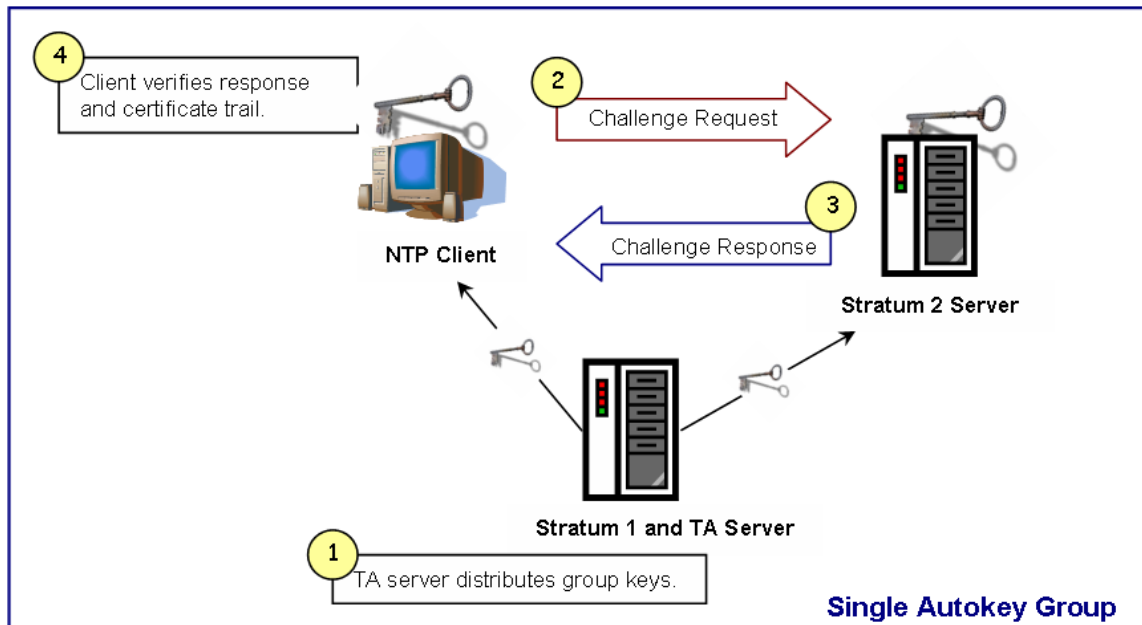
NTP includes provisions to restrict access to time servers by using an IP address or mask-based restriction list. The access control command *restrict* allows the server to set both time service as well as informational query information restrictions. This also includes restrictions for run-time modifications of the server configuration and packets that do not support the current NTP version.

### 6.14.3 Authentication

NTP includes provisions for ensuring authenticated clock synchronization using the *Autokey* protocol to prevent masquerade and middleman attacks. NTP version 4 supports symmetric key and public key algorithms to verify message and sequence number integrity to prevent replay attacks. NTP version 4 also supports digital signatures, X.509 certificates, and several identity verification schemes based on cryptographic challenge-response algorithms to authenticate the source of the packets. The Autokey protocol is based on the OpenSSL library's public key infrastructure (PKI) algorithms. The public key management can either be provided by commercial services or produced by utility programs in the OpenSSL library. Three variants of the Autokey protocol allow the security features to function with any of the association modes.

The Autokey protocol is enabled on an NTP node by default; however, it is ineffective unless it is properly configured. Authentication can be configured using the *key* or *autokey* command for each *peer*, *server*, *broadcast*, and *manycastclient* assignment in the configuration file. The *autokey* command specifies the location of the key files, which symmetric keys are trusted, and the interval between key refreshes. When no values are assigned to the command, the default values are used. The default refresh rate value is 12 or about every 1.1 hours. The node will check for the encrypted authentication fields. The *auth* flag ensures new associations, and remote configurations must be authenticated. If the flag is disabled, new association modes such as a broadcast server can potentially disrupt the client's timing (Figure 8).

How each exchange takes place is determined by the authentication mechanism and the challenge-response identity scheme used. The Time Authority server is typically a stratum 1 server, but is not restricted to stratum 1. Also, the Time Authority server does not have to be an NTP host.



**Figure 8 Autokey Protocol Exchange**

## 6.15 Equipment Internal Time Synchronization

The equipment is responsible for synchronizing the internal components that require precise timing. Synchronization of internal clocks should be application-driven, due to design challenges to optimize performance and minimize cost [15]. The following describes the design guidelines for internal time synchronization.

### 6.15.1 Robust

*Timing errors due to failures in equipment clocks should be contained.*

**Implementer:** Equipment Suppliers

**User:** Device makers

**Standards:** SEMI E148

#### Remarks:

Internal equipment time synchronization should be able to sustain the failure of other system components, such as individual sensors. If there is a single master clock and the clock fails, the problem may cascade to all clocks in the system. If unexpected or periodic failures do occur, the errors in the timing of a single clock should not be propagated to the other clocks in the system. Ideally, the synchronization of components should not depend solely on the factory time server due to potential unavailability.

### 6.15.2 Resource Aware

*Internal time synchronization protocols should be simplified.*

**Implementer:** Equipment Suppliers

**User:** Device makers

**Standards:** SEMI E148

**Remarks:**

Resource limitations in the equipment components drive the need to simplify internal time synchronization protocols. Embedded components, including sensors, are often limited in processing capacity and memory. Integrating the time synchronization protocol into the hardware may allow the tighter integration needed to reduce delays and variations in processing. Internal synchronization should minimize the number of messages, while rapidly synchronizing the components.

### 6.15.3 Configurable

*Users should be able to enable or disable the time synchronization of components.*

**Implementer:** Equipment Suppliers

**User:** Device makers

**Standards:** SEMI E148

**Remarks:**

Some applications require periodic time synchronization rather than synchronization at all times. To reserve system resources, it may be beneficial to allow both constant synchronization as well as periodic or on-demand synchronization.

## 6.16 Data Time-Stamping Guidelines

Following are time-stamping related guidelines. Epoch, timestamp format, and time-stamping data are defined.

### 6.16.1 Time Base

*The use of NTP epoch is recommended for time-stamping.*

**Implementer:** Device makers

**User:** Device makers

**Standards:** SEMI E148

**Remarks:**

- Because it is presumed most systems will rely on NTP, use of the NTP epoch is recommended as the time base for system clocks.

**Background:** The epoch is the reference date based on an event for measuring the passing of time. Some common epochs are the NTP epoch, which began on January 1, 1900, while the GPS epoch began January 6, 1980. A consistent epoch among each system ensures that the timestamps are consistent and eliminates further calculations and complications when processing the timestamps for correlating data from different sources with different epochs. The basis of the timestamp does not necessarily have to begin at the start of an epoch. A timestamp basis starting at the beginning of a process may be sufficient for process diagnosis and will require less allocation of resources for timestamps (i.e., timestamp size). However, equipment fault diagnosis may take a longer time.

### 6.16.2 Format

*Time-stamping applications must use the ISO 8601 format as defined by SEMI E148.*

**Implementer:** Equipment and third-party software suppliers

**User:** Device makers

**Standards:** SEMI E148

**Background:** Many nations have different methods of representing the date and time. For instance, the U.S. customarily represents August 4, 2004, as 08/04/2004, which other countries may interpret as April 8, 2004. ISO 8601:2004 *Data Elements and Interchange Formats-Information Interchange – Representation of Dates and Times* [8] specifies a standard framework for numeric representation of date and time to avoid confusion in international communication, especially for increasing the portability of computer communication applications. Time representation according to ISO 8601 has already been adopted internationally. The standard does not require compliance with a specific format, but rather defines a framework for representing date and time [6].

### 6.16.3 Time-Stamping

*Timestamps coupled with a piece of data must reflect the time the data was measured or generated as closely as possible to provide a sufficient level of data quality.*

**Implementer:** Equipment and third-party software suppliers

**User:** Device makers

**Standards:** SEMI E148

**Remarks:**

- Including accuracy information on the timestamp, such as estimated clock deviation from UTC and estimated time-stamping latency from when the event occurred to when it was actually time-stamped, is recommended.
- Ideally, timestamps should not include the time a piece of data is in a transmission or processing queue or any data transmission or processing time.

- If timestamp latency is inevitable, the time-stamping application should make provisions to estimate or calculate any significant latency. One method to mitigate the latency is to estimate and subtract the latency from the actual timestamp to generate an estimated timestamp.

#### 6.16.4 Time-Stamping Resolution

*Time-stamping resolution should reflect the resolution of the clock source.*

**Implementer:** Equipment suppliers, software suppliers

**User:** Device makers

**Standards:** SEMI E148

**Remarks:**

- The resolution of the timestamp should not be greater than the accuracy of the clock generating the timestamp. The remaining places set aside for the timestamp should be 0s to reflect only the available clock accuracy.
- Applications generating timestamps may also be limited by the programming language in which they are written.
- Samples of default operating system clock tick frequency are as follows:
  - Windows: Approximately once every 10–15 ms by default. Methods exist to improve the timer resolution to 1 ms [12].
  - Linux: Approximately once every 1 ms. The kernel can be modified for lower resolution. FreeBSD uses the NTP nanokernel, which currently has microsecond resolution but strives to achieve nanosecond resolution.
  - UNIX: Approximately once every 1 ms. The kernel can be modified for lower resolution.
  - Solaris: Approximately once every 1 ms. The kernel can be modified for lower resolution.
  - QNX (Real-time operating system): Depending on the system clock, the resolution can be a number of nanoseconds. The exact resolution can be determined by using the *ClockPeriod()* call. Clock cycles can be determined by CPU cycles, which are approximately 10 ns [10].

## **APPENDIX A – ACCURACY/PRECISION GUIDELINES FOR e-MANUFACTURING APPLICATIONS**

The following time synchronization requirements tables reflect the need for time synchronization across a wide variety of semiconductor e-Manufacturing applications. These requirements are based on the collective experience and perspective of the Time Synchronization Working Group, which includes representatives from device makers, software suppliers, equipment suppliers, universities, and research consortia. The data compiled do not necessarily indicate the most stringent requirements; rather, they represent what is considered sufficient to support semiconductor production facilities between now and 2010.

These requirements arise because manufacturing processes happen in real-time and the applications that manage and control them are highly distributed (across the factory, across the company, and across a variety of computing platforms). Coordinating these applications frequently depends on knowing “what time it is”; likewise, analyzing and interpreting their results often requires knowing “what time it was.”

There are several distinct audiences for these guidelines. First, standards developers are now looking at how time is handled in the current generation of standards to fill the gaps with modifications and/or new specifications. Factory systems engineers must understand the breadth of these requirements to define system architectures, select protocols, and establish procedures that can maintain the necessary degree of synchronization across the affected environments. Finally, application engineers (from suppliers of equipment, software, subsystems, or internal custom applications) can use this information to ensure their programs are good citizens of an overall, coordinated system.

The guidelines tables are organized by “environment,” which in this context means some layer or subset of a factory’s information and control system within which several applications are directly related and may be interdependent. The environments include on-tool (i.e., a single piece of equipment); a single process area (group of identical/similar tools); a process module, or group of related (perhaps even interconnected); the overall factory or manufacturing site; and the enterprise (single company or manufacturing partnership).

For each application, four timing factors were considered. “Absolute Accuracy” means how close to absolute time a particular value must be (as defined by the designated global time authority [i.e., UTC]). “Relative Accuracy” is how close to some reference time within the same environment a particular value must be (i.e., how much drift is allowed before a resync is needed). Further definitions for absolute accuracy and relative accuracy are in Section 4. “Minimum Data Sampling Interval” refers to how often each value (or buffer of values) is sampled by the application. “Precision Required” is the degree of discrimination with which a time quantity is stated (also synonymous with “resolution,” see Section 4). Based on each application’s features, dependencies, and interactions, values were estimated from end-user and equipment supplier experiences as well as discussions in the working group. The consensus results are captured in Table A-1 through Table A-5.

**Table A-1 On-Tool Application – Single Piece of Equipment Context**

<b>On-Tool Application</b>	<b>Description/Needs</b>	<b>Absolute Accuracy (sec)</b>	<b>Relative Accuracy (ms)</b>	<b>Minimum Data Sampling Interval</b>	<b>Precision Required</b>
Real-Time Tool Control	Ability to collect data for real-time feedback control. Ability to guarantee tool will respond to command within a specified period of time.	5	1	1 ms	0.1 ms
In-Situ FDC	Need to correlate sensor data, equipment data, and other process data to a specific wafer/lot for multivariate analysis.	5	5	5 ms	1 ms
Integrated Metrology (IM)	Standard calls for synchronized clocks to accurately timestamp metrology data.	5	100	100 ms	1 ms
Tool-side Interface A Implementation	Need data at higher sampling rates and with more consistency. All data items must have timestamps.	5	10	Per-tool sampling interval 10 ms In aggregate, a tool must handle 10,000 variables per second	1 ms
e-Diagnostics Support	Provide secure, tool state-dependent data collection and communication to external applications (typically a local e-Diagnostics server).	5	5 ms	5 sec (remote data collection requirement) 10 ms (local data collection requirement)	1 ms
Stepper/Scanner		5	1	50 ms	10 us

**Table A-2 Close to the Equipment Applications – Group of Identical/Similar Tools**

Close to the Tool or Module Application	Description/Needs	Absolute Accuracy (sec)	Relative Accuracy	Minimum Data Sampling Interval	Precision Required
Arc Detection	Analyze power supply and other sensors to determine conditions that lead to arcing; can also monitor these conditions to prevent arcs.	5	200 us	500 us	100 us
Equipment/ Chamber Matching	Calculate and compensate for variations between sets of "identical" processing elements (tools, chambers).	5	10 ms	10 sec	1 ms
Equipment Integration (Host-side Interface A and GEM/SECS Implementations)	Need data at higher sampling rates and with more consistency. All pieces of data must have timestamps.  Increasing use of adaptive sampling.	5	5 ms	Handle trace data collection at 20–50 Hz per variable.  Equipment must support total of at least 10,000 variables per second.	1 ms
Statistical Process Control (SPC)	Track equipment/process parameters and flag SPC rule violations; not a real-time control technique.	5	100 ms	1 sec	1 ms

**Table A-3 Factory/Process Equipment Application – Group Related Tools**

Factory/Process Level Application	Description/Needs	Absolute Accuracy (sec)	Relative Accuracy	Minimum Data Sampling Interval	Precision Required
Run-to-Run Control (R2R)	Adjust recipe parameters between individual production runs (wafer, lot, batch) based on results of previous runs; uses combination of metrology and equipment data to calculate adjustments.	5	50 ms	100 ms	1 ms
Fault Detection and Classification (FDC)	Analyze equipment and process parameters to ensure equipment is in its acceptable operating envelope; identify and classify (or prevent) equipment faults and interrupt processing accordingly.	5	10 ms	50 ms (in process) 20 sec (post-process)	1 ms
Event/Alarm Management	Capture, analyze, communicate, and support user response to events and alarms across the production environment.	5	1 sec	1 sec	1 ms
Yield Management/ Engineering Data Analysis (YMS, EAS)	Correlation of information from broad variety of equipment, engineering, failure analysis, and production datasets.	10	50 ms	1 min	1 ms

**Table A-4 Factory-Level Application – Factory Logistics**

<b>Factory Level Application</b>	<b>Description/Needs</b>	<b>Absolute Accuracy (sec)</b>	<b>Relative Accuracy</b>	<b>Minimum Data Sampling Interval (sec)</b>	<b>Precision Required (ms)</b>
Scheduling/ Dispatching (RTD)	Provides ability to accurately estimate time of completion and arrival of wafers and to prepare a piece of equipment for wafer processing.	10	2 sec	2	1
Automated Material Handling (MCS, AMHS)	Transport and store production material and fixtures.	10	500 ms	1	1
Network Security	Authentication systems must have accurate timestamps to within minutes; however, audit trails may have more stringent requirements.	10	1 sec	2	1
Factory Time Synchronization	Maintain and provide reference time for all systems in the factory.	5	1 ms	10	1

**Table A-5 Enterprise-Level Application – Single Company**

<b>Enterprise Level Application</b>	<b>Description/Needs</b>	<b>Absolute Accuracy (sec)</b>	<b>Relative Accuracy (sec)</b>	<b>Minimum Data Sampling Interval</b>	<b>Precision Required (ms)</b>
Remote Production Monitoring (e.g., Dashboard)	Flexibly present current and historical production information to stakeholders regardless of location	30	2	5 sec	1
Enterprise Resource Planning (ERP)	Set of applications that automate order processing, manufacturing scheduling, customer shipping, and related finance and human resource functions	30	30	2 min	1



**International SEMATECH Manufacturing Initiative  
Technology Transfer  
2706 Montopolis Drive  
Austin, TX 78741**

**<http://ismi.sematech.org>  
e-mail: [info@sematech.org](mailto:info@sematech.org)**