

**NISTIR 7184**

**Semiconductor Factory and Equipment Clock  
Synchronization for e-Manufacturing**

Ya-Shian Li  
Brad Van Eck

**NIST**

**National Institute of Standards and Technology**  
Technology Administration, U.S. Department of Commerce

**Semiconductor Factory and Equipment Clock  
Synchronization for e-Manufacturing**

**Ya-Shian Li**

*Semiconductor Electronics Division  
Electronics and Electrical Engineering Laboratory  
National Institute of Standards and Technology  
Gaithersburg, MD 20899-8530*

**Brad Van Eck**

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

December 2004



**U.S. DEPARTMENT OF COMMERCE**  
*Donald L. Evans, Secretary*  
**TECHNOLOGY ADMINISTRATION**  
*Phillip J. Bond, Under Secretary of Commerce for Technology*  
**NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY**  
*Arden L. Bement, Jr., Director*

## **Abstract**

This study describes the potential methods, challenges, and recommendations for addressing factory and equipment clock synchronization and time stamping. The report also discusses why these are critical to the industry as the upcoming generations of tools and fabrication facilities are being designed and constructed. It will also recommend areas for standardization and best practices for clock synchronization and time stamping.

## **Keywords:**

Automated Data Collection, distributed systems, IEEE 1588, Network Time Protocol, time stamping, time synchronization

## **Abbreviations:**

*AEC* – Advanced Equipment Control  
*APC* – Advanced Process Control  
*CPU* – Central Processing Unit  
*FDC* – Fault Detection Classification  
*GPS* – Global Positioning System  
*LAN* – Local Area Network  
*NTP/SNTP* – Network Time Protocol/Simple Network Time Protocol  
*OEM* – Original Equipment Manufacturer  
*PTP* – Precision Time Protocol (IEEE 1588)  
*TCP/IP* – Transmission Control Protocol/Internet Protocol  
*UTC* – Coordinated Universal Time  
*UDP/IP* – User Datagram Protocol/Internet Protocol  
*WAN* – Wide Area Network

## **Disclaimer**

*Company names and commercial products mentioned within this document are for information only; it does not imply recommendation or endorsement by NIST.*

## TABLE OF CONTENTS

<b>1 EXECUTIVE SUMMARY</b> .....	<b>1</b>
<b>1.1 PROBLEM</b> .....	1
<b>1.2 POTENTIAL SOLUTIONS</b> .....	1
<b>1.3 RECOMMENDATIONS</b> .....	2
<b>2 INTRODUCTION</b> .....	<b>2</b>
<b>3 THE USUAL SUSPECTS: SOURCES OF TIME FLUCTUATION IN NETWORKS</b> .....	<b>4</b>
<b>3.1 NETWORK TRAFFIC</b> .....	4
<b>3.2 NETWORKING DEVICES</b> .....	5
3.2.1 <i>Repeaters</i> .....	5
3.2.2 <i>Switches</i> .....	5
3.2.3 <i>Routers</i> .....	5
<b>3.3 OPERATING SYSTEM</b> .....	5
<b>3.4 CPU LOAD</b> .....	6
<b>3.5 OSCILLATOR</b> .....	6
<b>4 NEEDS FOR PRECISION TIME IN OTHER INDUSTRIES</b> .....	<b>6</b>
<b>4.1 ELECTRIC POWER INDUSTRY</b> .....	6
<b>4.2 AUTOMOTIVE INDUSTRY</b> .....	6
<b>4.3 PHARMACEUTICALS</b> .....	7
<b>4.4 PAPER INDUSTRY</b> .....	7
<b>4.5 FINANCIAL SERVICES</b> .....	7
<b>4.6 COMPUTER FORENSICS</b> .....	7
<b>4.7 TELECOMMUNICATIONS</b> .....	7
<b>5 SEMICONDUCTOR INDUSTRY NEEDS FOR PRECISION TIME</b> .....	<b>9</b>
<b>5.1 CHIP MANUFACTURERS PERSPECTIVE</b> .....	9
<b>5.2 EQUIPMENT MANUFACTURERS PERSPECTIVE</b> .....	10
<b>5.3 MANUFACTURING SOFTWARE PERSPECTIVE</b> .....	10
<b>6 POTENTIAL SOLUTIONS</b> .....	<b>11</b>
<b>6.1 TIME SYNCHRONIZATION PROTOCOLS FOR DISTRIBUTED SYSTEMS</b> .....	11
<b>6.2 TIME SOURCES</b> .....	11
6.2.1 <i>Atomic Clock</i> .....	11
6.2.2 <i>GPS Receivers</i> .....	12
6.2.3 <i>Radio Receivers</i> .....	12
6.2.4 <i>Internet</i> .....	12
<b>6.3 PRECISION OSCILLATORS</b> .....	12
6.3.1 <i>Quartz Oscillators</i> .....	13
6.3.2 <i>Atomic Oscillators</i> .....	13
<b>6.4 POTENTIAL SOLUTIONS FOR THE SEMICONDUCTOR INDUSTRY</b> .....	14
<b>7 NETWORK TIME PROTOCOL (NTP)</b> .....	<b>14</b>
<b>7.1 ARCHITECTURE</b> .....	14
<b>7.2 ALGORITHM</b> .....	15
7.2.1 <i>Data Grooming Algorithms</i> .....	15
7.2.2 <i>Clock Discipline Algorithm</i> .....	15
<b>8 SIMPLE NETWORK TIME PROTOCOL (SNTP)</b> .....	<b>16</b>
<b>9 IEEE 1588</b> .....	<b>16</b>

<b>9.1 ARCHITECTURE</b> .....	<b>16</b>
9.1.1 <i>Boundary Clock</i> .....	17
9.1.2 <i>Node Hardware</i> .....	17
<b>10 TIME SYNCHRONIZATION RECOMMENDATIONS</b> .....	<b>18</b>
<b>10.2 NETWORK SYNCHRONIZATION PROTOCOL CONFIGURATION</b> .....	19
<b>10.3 RECOMMENDATIONS FOR SELECTING A PROTOCOL</b> .....	19
<b>10.4 NETWORK TOPOLOGY RECOMMENDATIONS</b> .....	22
<b>10.5 EQUIPMENT TOPOLOGY RECOMMENDATIONS</b> .....	22
<b>11 TIME STAMPING</b> .....	<b>22</b>
<b>11.1 STANDARD TIME</b> .....	22
<b>11.2 TIME REPRESENTATION</b> .....	23
<b>11.3 EPOCH</b> .....	23
<b>12 POTENTIAL STANDARDS EFFORTS</b> .....	<b>24</b>
<b>13 CONCLUSIONS</b> .....	<b>25</b>
<b>13.1 ACCURACY AND PRECISION REQUIREMENTS</b> .....	25
<b>13.2 COST</b> .....	25
<b>13.3 NEXT STEPS</b> .....	25
<b>14 REFERENCES</b> .....	<b>26</b>

## LIST OF FIGURES

Figure 1	Potential Causes of Latency and Other Degradation in Distributed Time Synchronization Performance. ....	4
Figure 2	Simplified IEEE 1588 Architecture with Two Subnets. ....	17
Figure 3	Co-Existence of NTP/SNTP and IEEE 1588 in a Single Network. ....	20
Figure 4	IEEE 1588 as the Primary Protocol. ....	20
Figure 5	Possible Time Stamping Points. ....	24

## **1 EXECUTIVE SUMMARY**

Increased wafer sizes, decreased critical dimensions, and new materials all challenge higher initial yield, lower manufacturing costs, and time-to-market demands in the semiconductor industry. Among the key issues driving these challenges are the monitoring and maintenance of wafer surface uniformity, decreasing process tolerance windows, increasing number of factors affecting yield, and increasing process and equipment complexity. Solving these issues will require data to be collected and analyzed from the process equipment, in-line metrology, and factory environmental conditions. Time stamps provide a scalable, efficient, and technically sound mechanism for merging information retrieved from the rising confluence of heterogeneous data streams. Accurate clock synchronization and time stamps are fundamental to the e-Manufacturing paradigms such as Advanced Process Control (APC), e-Diagnostics, real-time control, and development of an intelligent automated knowledge base for rapid yield learning. Precise clock synchronization will also enable more intelligent and efficient automation with better coordination between hardware and software systems. Tools capable of accurately estimating and propagating their completion time to coordinate with the arrival of the next set of wafers to process can reduce the tool waiting time.

### **1.1 Problem**

Network time synchronization is an advantageous, if not critical, component in real-time automation. Networked-based automation systems typically require synchronized clocks with better coordination to be more efficient and less prone to errors. Intelligent data analysis also requires accurate time stamps to correctly merge data from heterogeneous sources. Synchronizing the systems in a distributed environment is not a trivial issue. Many elements in the end systems, networking devices, and network can degrade synchronization performance.

The semiconductor industry has been plagued with unreliable, and at worst unavailable, time information. The critical cause can be attributed to the inability to distribute accurate time to the clocks within the tool, where suppliers have observed time stamp latencies of up to *2 minutes*.

### **1.2 Potential Solutions**

Time synchronization issues are inherently similar; therefore, the solutions developed are applicable to a wide range of industries spanning from telecommunications to automotive. Cross-industry solutions are ever-evolving, as the timing synchronization requirements become increasingly stringent. While each industry has its own specific timing requirements, there are a few common components among the solutions. Maintaining precise, accurately synchronized time in a distributed environment requires a solution comprised of a quality reference time source, a synchronization protocol for distributing time information to the network nodes, software to process the timing information, and a precision clock at each node requiring highly precise time accuracy and stability.

The international standard time (Coordinated Universal Time [UTC]) provides the most accurate and stable time reference. It can be best acquired by a global positioning satellite (GPS) receiver. Network Time Protocol (NTP) and Institute of Electrical and Electronics Engineers (IEEE) 1588 are two solutions designed for distributing UTC time throughout the network. Optimizing time synchronization requires the equipment, network, and communication paradigm to be amenable to leveraging the synchronization protocols. NTP is a time-tested, evolving synchronization protocol that has been widely used in Internet and enterprise local area network (LAN) environments for accuracies on the order of 1 millisecond (ms) in a LAN environment synchronized to a GPS time source. As an internationally recognized solution adopted by American National Standards Institute (ANSI) and International Electrotechnical Commission (IEC) (IEC 61588), IEEE 1588 supports sub-millisecond synchronization accuracy requirements for next-generation industrial automation. In addition to time distribution, individual nodes must be able to maintain accurate time to avoid frequent synchronizations. Precision oscillators for the equipment host and even end nodes are also available to meet industrial timing requirements.

### 1.3 Recommendations

With NTP as the current prevailing solution for disseminating time information, the industry has a time synchronization protocol sufficient to meet near-term accuracy demands. Meeting more demanding and complex automation needs will require sub-millisecond accuracies. The industry should continue to follow and even actively participate in the development of IEEE 1588 to incorporate the capabilities into next-generation equipment, network, and other system designs for meeting industry requirements.

High-end factory networks with message latencies of less than 1 ms are sufficient to support available time synchronization protocols to their maximum capability. Therefore initial industry efforts should be focused on the equipment. The industry should amass experts to draft guidelines and standards to facilitate the communication of timing information between equipment and within a single tool. Equipment suppliers should also be aware of best practices for time synchronization and time stamping when designing future processing and metrology tools.

Developing standards will reduce the cost, maintenance, and integration time of factory-wide time synchronization and time-stamping solutions. Current message communication standards, such as the sensor bus, should support the transmission of NTP and IEEE 1588 timing information. System clocks should be able to propagate their current time and worst-case precision and accuracy in a standard format upon request. Standard time stamp formats and reference epochs should avoid the need to translate between heterogeneous systems. For quality data, a format for conveying time stamp quality may also need to be developed. The time stamp quality can be judged by the accuracy and precision of the clock as well as the point where the time stamp was generated. For optimal time stamp quality, the time stamp should be generated by a synchronized precision oscillator at the exact time the data was generated.

With abundant benefits to reap, applicable solutions available and minimal long-term management needed, it becomes evident the industry should view accurate time synchronization and time stamping as a critical component in their factory networks.

## 2 INTRODUCTION

In any network-based computing environment where software and hardware must interact for proper and efficient function or data correlation, shared applications typically require accurate clock synchronization and time stamps. Whether the shared systems reside in a single building or span global sites, whether they are motion control systems or simple personal computers (PCs), the single concept they share simultaneously is time. Time, therefore, plays a significant role in both *event synchronization* and *data correlation*. The first step to understanding time synchronization requirements of an organization's network is to determine how interacting applications in the organization can benefit from accurate time.

For event synchronization, temporal information is typically required to ensure events occur on time, in the correct sequence, for a specified duration, or a combination of the above. Examples are abundant in many manufacturing industries including the pharmaceutical industry, where materials must be processed in batches for a specified period of time. Processing a batch for too long or not long enough could render the final product ineffective or even harmful.

Accurate time stamps, a necessary component in ensuring data quality for correlation analysis, also require accurate clock synchronization. Relative to time, *accuracy* describes the offset between a source time (i.e., UTC) and the time on the local clock. Time stamps are often used for forensic investigations in the digital age, computer security, and fault diagnosis. One of the more highly publicized needs for accurate time stamps was made evident while trying to diagnose the cause of the Northeastern blackout of August 2003. Because the time stamps were not accurate enough to simply compare the events from the distributors throughout the grid, the



investigators had to recreate the actual time index to pinpoint the root cause by crosschecking time stamps, thereby delaying the findings for weeks.

It has become evident that many applications, ranging from the mundane to cutting-edge automation, require accurate time. Simply maintaining network operations, such as enterprise-wide network security, fault diagnosis, and usage auditing, requires accurate time stamps on the order of milliseconds. In industrial environments, real-time control among heterogeneous software and automation equipment inherently requires sufficient synchronization. *Real-time* capability refers to the ability to guarantee an action or response *within a specified time*. Any application sensitive to the timeliness of a signal will require clocks to be synchronized among the interacting systems. Real-time applications can also require time stamps. Real-time actions dependent on dynamic data analysis play a significant role in process control, a manufacturing process technology intended to fully exploit automation capabilities.

Within a semiconductor manufacturing facility, several applications from the office to the factory floor require network time synchronization. Intuitively, any application requiring diagnosis of cause and effect when several heterogeneous systems are involved will require time stamps. Time becomes the traceable, common factor among the disparate data streams. Some key applications include fault detection classification (FDC) and e-Diagnostics. Any form of APC and advanced equipment control (AEC) where an electronic system controls the duration or synchronization of a process requires accurate, synchronized clocks. Any motion control system requires synchronization with both control software and any complementing equipment to prevent unscheduled waiting time.

In an e-Manufacturing environment, both hardware and software must be able to obtain accurate, precise time to function properly or to maximize the benefits of real-time e-Manufacturing. So why should the industry devote its limited resources to address clock synchronization and time stamping issues *now*, when the tradeoff between consequences of inaccurate time stamps and the potential cost of implementing a reliable time-stamping infrastructure is still ambiguous? Some key motivations include the low yields at the 90 nm process node and advancement of resource efficient automation at 300 mm. Increasing factory efficiency has driven standard equipment communication interfaces, such as Interface A, which are being developed and deployed to promote data availability. In the mean time, the industry should be prepared to efficiently manage and use the influx of data.

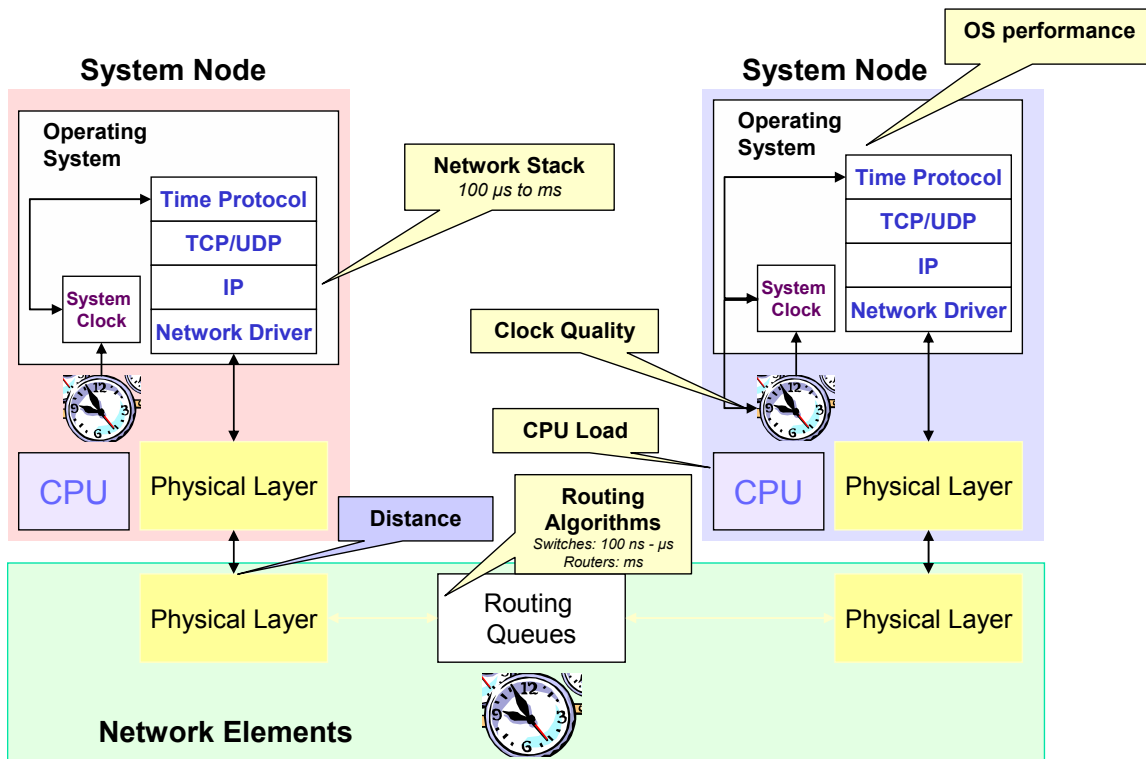
Improving the time stamping of factory systems is not easy. Plagued with incompatible equipment architectures and inconsistent data formats, the industry should begin addressing issues and developing a technically robust system amenable to accurate time synchronization and time stamping. The critical efforts required to achieve these goals lie in the process and metrology equipment architectures, which have been, at best, inconsistent for supporting accurate synchronization of its system clocks. One equipment supplier has observed current time-stamping latencies ranging from 100 ms to 2 minutes within a single piece of equipment.

Establishing a sound time synchronization and time-stamping system requires an understanding of methods to obtain and distribute accurate time within the equipment and throughout the factory network. Evolving over 20 years, NTP provides a means to synchronize clocks down to millisecond accuracies using a pure software implementation over existing standard network components. The industry should also be attuned to the developments in the industrial-based IEEE 1588 Precision Time Protocol (PTP) (i.e., IEEE 1588) for meeting more stringent accuracy demands. IEEE 1588 provides sub-millisecond accuracies, but achieving optimal accuracy levels requires IEEE 1588-based boundary clocks at the routers and specialized network interface cards at the clock nodes. Understanding the tradeoffs inherent in the selection of a specific time synchronization protocol as well as the tradeoffs between accuracy requirements and implementation complexity will also facilitate the development of a robust, cost-effective solution. This document describes the challenges of network-based synchronization, various network and

industrial applications, and more specific semiconductor industry needs, potential solutions, recommendations, and standards efforts.

### 3 THE USUAL SUSPECTS: SOURCES OF TIME FLUCTUATION IN NETWORKS

Before delving into why the semiconductor industry requires time synchronization in their factory systems, it is necessary to understand why electronic systems tend to have unstable clocks and some of the challenges of synchronizing clocks in a networked environment. Typical nodes in a distributed system are equipped with the cheapest clocks available and thus often the least stable, requiring frequent synchronizations to “discipline” the clocks. PC clocks can wander minutes per day depending on the load of the CPU, the operating system, etc. In addition to oscillator issues, the accuracy achieved by synchronization protocols depends on network traffic, networking devices, and any related system hardware/software. See Figure 1 for potential causes of time fluctuations. While the following discussion will primarily be based on Ethernet networks, similar issues can be applied to other packet-based networks.



**Figure 1** Potential Factors Causing Latency and Other Degradation in Distributed Time Synchronization Performance.

#### 3.1 Network Traffic

Typical synchronization protocols rely on symmetrical delays in calculating network latencies. For time synchronization protocols, predictable one-way latencies do not degrade synchronization as much as the variation in the delays. *Jitter* is the maximum variation of one-way latencies in the network. Reducing such latency variability requires sufficient network capacity and management to prevent congestion. In Ethernet, a non-deterministic protocol, 30% load on the total bandwidth is considered heavy traffic. A typical Ethernet network sustains loads from 5–30%. Jitter depends on network load, drop link speed, packet sizes, and network architecture. The *variability* in the transport latencies between two synchronizing nodes is among the critical factors degrading synchronization performance.

## 3.2 Networking Devices

In many Ethernet-based network environments, the interconnecting devices contribute to the non-deterministic nature of the system. Jitter is an inherent characteristic of non-deterministic Ethernet system. Reducing jitter in network traffic means removing or mitigating the factors causing the *variable* latencies. Interconnecting network devices can introduce significant variable latencies and, therefore, degrade time synchronization accuracy for any pure software-based protocol.

### 3.2.1 Repeaters

*Network repeaters* are passive devices that forward digital signals. Packets transported through a repeater are broadcasted to every node on the repeater without filtering. While the packets are being transmitted to nodes to which they are not addressed, causing unnecessary network traffic, the lack of processing allows the repeater to quickly forward packets with minimal delay. Repeaters also do not use the store and forward techniques that are characteristic of other network connector devices. The lack of queues ensures there are no buffering delays. Despite the minimized delays, repeaters are generally becoming obsolete since they often compromise overall network performance.

### 3.2.2 Switches

Synchronization packets traveling through switches incur latencies ranging from tens to hundreds of microseconds because of processing and store and forward techniques. Partial parsing of the packet header occurs at the switches since a packet is sent to the exact address specified. Network connections separated by switches are also in separate Ethernet collision domains. Therefore, the packets can reside indefinitely in a queue to prevent collisions during heavier loads.

### 3.2.3 Routers

Routers typically are used to link separate subnets either between buildings or to separate large networks residing in a single building. Routing algorithms rely on routing tables to queue network packets for ensuring optimal network performance given the present load. Ironically, the routing algorithms tend to degrade the performance of time synchronization protocols. The router adds to the uncertainty of symmetric delays since the buffering times are indefinite depending on network traffic. Packets are forwarded almost instantaneously if the network load is light enough, while heavier loads could delay a packet on the order of milliseconds.

## 3.3 Operating System

The operating system kernel is responsible for providing steady, periodic interrupts to advance time for software clocks. Typically, operating systems assign a fairly low priority to timing interrupts, since they are not considered a critical application. However, software-based time synchronization protocols can significantly improve with modifications to the interrupt service routines (ISRs) in the kernel. Assigning a higher priority for timing interrupts further improves clock accuracy, since fewer interrupts will be missed because they were preempted by higher priority applications. Operating systems are also typically the limiting factors in the precision and resolution of the system time and time stamps.

Real-time operating systems, such as VxWorks, offer more determinism by guaranteeing worst-case interrupt latencies, task switching, and process execution times. High priority tasks have a worst-case lockout time on shared resources. However, the ability to guarantee the latencies requires subsystem clocks to have sufficient resolution and to be disciplined and synchronized to ensure the worst-case waiting time is accurate; otherwise, a faulty timekeeping mechanism can prevent the operating system scheduler from running a higher priority task with critical deadlines. When deadlines are missed, systems may fail. The system clock should be able to be set at high frequencies to attain highly accurate task scheduling. Higher frequencies allow time slices to have shorter intervals and ensure that high priority tasks preempt current, lower priority tasks more rapidly.

### 3.4 CPU Load

The processing events on a system may also affect the timekeeping mechanism in a computing environment. If an application relies on a software clock, typical systems tend to delay or even neglect clock interrupts, therefore preventing time from advancing as the system simultaneously handles transmission, sensing, and error management tasks [1].

### 3.5 Oscillator

Cost-sensitive systems, such as typical computer clocks, are not precise and tend to wander after some period of time. Such inconsistencies are due to environment- and event-related reasons. Oscillator frequency can be affected by ambient temperature, pressure, supply voltage, vibrations, and humidity [2]. Even with frequent network synchronization, poor clocks can limit the achievable accuracy primarily because of their instability. In addition, more frequent synchronization is required, expending more network and system resources.

## 4 NEEDS FOR PRECISION TIME IN OTHER INDUSTRIES

While the semiconductor industry certainly has its own unique set of issues, when tackling the time synchronization and time stamping challenges, the accurate timing needs of other industries may provide some insight into applications requiring a common sense of time in their networked manufacturing systems. Each industry typically has several applications requiring accurately synchronized time for providing both quality data, driving system automation, and maintaining network operations.

### 4.1 Electric Power Industry

The electric power industry must synchronize the phase to deliver power at the specified hertz (typically 60 Hz or 50 Hz). Generators that are not synchronized could harm other generators. Frequency is typically maintained by a stable source such as the atomic time.

The industry further relies on time synchronization and time stamps to monitor the power grid for events such as voltage spikes or equipment outages. Supervisory control and data acquisition (SCADA) systems rely on GPS time servers to synchronize clocks and provide accurate time stamps. Power grid substation and metering automation require accuracy from 1 ms to 1 second.

### 4.2 Automotive Industry

The automotive manufacturing factory relies on continuous flow, robotic production lines to fabricate vehicles. Robotics is pervasively used for automated assembly, welding, painting, etc. Automotive manufacturers are continuously seeking methods to expedite return on assets (ROA). Manufacturers push for speed until production quality is compromised or process bottlenecks occur. However, even with aggressive manufacturing techniques, factories fail to optimize productivity. The highest productivity is seen only when each operational section is synchronized with a previous station to eliminate unscheduled downtime. Inter-device coordination, such as the ability of a robot to perform a task without a part coming to a complete halt, increases throughput significantly. Coordinating robotics and subassemblies such as the conveyors around the operating zone requires synchronization to provide *continuous flow* and eliminate start and halt operations that increase the average manufacturing line speed.

The trend towards modular equipment architectures has challenged the capability to maintain a common sense of time within the equipment. Equipment architecture is becoming more modular to use the best available technology for increased throughput. The trend towards off-the-shelf modular architecture requires better communication mechanisms than the traditional proprietary data links with a centralized control system. In addition, better production controls are required because of the loss of centralized system control. One automobile manufacturer plans to resolve the issue by requiring the equipment manufacturers to support IEEE 1588 from an external source and by propagating the accurate time reference to the subcomponents by SNTP.

### **4.3 Pharmaceuticals**

The primary need for time synchronization in the pharmaceutical industry lies in process control. For example, a recipe may call for a batch of materials to be heated to a certain temperature for an exact period of time and passed to another process for mixing. If a process begins before another is finished, the production equipment could also be damaged.

Due to strict regulatory control of pharmaceutical products, time stamping is required under FDA regulation 21 CFR Part 11. Recipes must be executed precisely with each step documented and time stamped to prove the batch adhered to a specific regulation. The regulations require accurate time stamping of all process data to ensure product safety, quality, and traceability to the root of a problem if one becomes evident.

### **4.4 Paper Industry**

The paper industry strives for low cost, high quality, and low variability in quality. While automation lowers costs in the long-term, the decrease of human intervention requires systems to be better synchronized. Quality data and time become critical factors in making decisions on actions. One application for time synchronization lies in the quality monitoring and control systems. The pulp and paper mill is characterized by a network of process units with complex interactions and numerous constraints. Similar to the semiconductor industry, the interactive nature of paper processing and refining is riddled with factors contributing to the final paper quality. With variable production rates, unscheduled shutdowns, etc., time synchronized data is essential to determine the effect of the mill operations on the final paper quality.

Precisely synchronized clocks enable quality time-stamped process control data, which is useful for data conditioning and data merging in their quality control systems. *Data conditioning* involves filtering suspect data before analysis. Time plots serve as a useful tool during this process. The more accurate the time stamps, the better the elimination of unrepresentative data. The *data merging* process locates and provides the variables used in quality analysis. Data merging is often complicated because the data are being generated at separate locations with different time stamps and different and inconsistent intervals ranging from 1 second to once an hour. Therefore, the first step is to select the time stamp basis for all of the related data, if the related systems do not have a uniform time stamp format and start time. The next step is to determine the time relationship of each of the variables. Without a common sense of time among the systems generating the data, determining the time relationships is a painstaking and time consuming process requiring detailed knowledge of the process. This detailed knowledge can be acquired only by generating more process data.

### **4.5 Financial Services**

In financial services, traders must deal with rapidly changing prices. The networks over which traders buy and sell must be synchronized to ensure the prices are still effective at the time of sale. Time stamps also provide a mechanism to prove the value of investments and transactions that take place at any particular moment. Accuracy requirements in this area typically range from 100 ms to 10 s.

### **4.6 Computer Forensics**

In the digital age, criminal indictments have also begun to rely on time-stamped log data to prove the occurrence of an illegal action. Legislation such as Sarbanes-Oxley and Health Insurance Portability and Accountability Act (HIPPA) require verifiable time stamps to prove the state and control of documents at particular points, including when they were created or altered and by whom.

### **4.7 Telecommunications**

One of the most stringent distributed clock synchronization demands is in the telecommunications industry. The major challenge is accurate *global* synchronization to emulate circuit-switching based on digitized packet-switched networks.

#### 4.7.1 Packet Synchronization/Circuit Emulation

In telecommunications applications, source information is split into messages or packets that are often routed independently to their destination. This means packets originating from the same source and heading for the same destination can be routed through different nodes in the network. Receiver equipment must be able to equalize the different delays of the received packets, thus rebuilding the original bit stream as if it were transmitted across a circuit-switched network. *Delay equalization* is achieved by establishing the original sequence through adaptive techniques or by the time stamps specified in the packet headers. Packets that fail to be ordered correctly can result in gaps during the conversation. Cell phone transmissions must provide the illusion of continuous analog signals by ensuring the time stamps are accurate at a level of a few microseconds.

It is advantageous in the telecommunications industry to distribute a common sense of time to every node over a global network to transmit and switch in an integrated digital format so that every network element can operate synchronously with the others and process all the incoming bit streams.

#### 4.7.2 Real-Time Synchronization

Real-time synchronization is used for network control and management. Events detected by the equipment monitoring system such as bit error rate threshold overflow, line alarms, hardware failures, etc., are recorded for future reporting. Management messages, which are sent to the operation system, must include key information such as *date and time*. Maintaining the requirements for management purposes requires only milliseconds of accuracy; however, synchronization of digital telecommunications networks typically achieves time deviations not greater than 10 to 100 nanoseconds (ns).

### 4.8 Real-Time Clock Synchronization

Any real-time system requires distribution of *absolute time* for network management and quality of service (QoS). Achieving real-time capabilities requires the ability to guarantee a service within a specific time range; therefore, every system in a real-time network must be synchronized.

#### 4.8.1 Multimedia Synchronization

Real-time multimedia synchronization, such as streaming media over the Internet, is a common application requiring synchronization. Orchestration of heterogeneous network elements (images, text, audio, video, etc.) in multimedia communication at different levels of integration (i.e., physical/human interface) requires time-stamped packets to accurately order the packets for quality playback.

### 4.9 Other Network Operations

Network and software performance testing also require the synchronization of computer systems so that performance data, such as processing time on a single node or transport time between computing nodes, can be accurately measured. Maintaining logs requires time stamps to accurately pinpoint the time failures, debug network issues, and provide a source of evidence for computer forensics applications. Such applications will further require a time stamp audit trail to ensure the time stamps are indeed accurate at a specific point in time.

Computer security applications such as digital certificates, digital signatures, and audit logs use time stamps as well. Encryption protocols typically use the time stamps and specify a temporal validity interval to prevent replay attacks. More accurate synchronization of related systems will permit shorter intervals and therefore enhance the security of an application. Authentication typically relies on a time stamp to generate a key. If the clocks are out of synch, the key generated by the sender according to its system clock that is running behind may have already "expired" according to the clock of the receiving system that has a more accurate time. Security policies requiring accurate time stamps must also ensure the integrity of the time synchronization protocol to avoid unauthorized access to the system time.

## 5 SEMICONDUCTOR INDUSTRY NEEDS FOR PRECISION TIME

Key applications for accurate time synchronization and time stamping in the semiconductor industry include merging heterogeneous data sources for improved data analysis in APC applications. Managing the data proliferation to extrapolate intelligent conclusions in a timely manner will be critical in preserving the integrity of the data and related applications. The exponential increase in data collection rates required to meet future technology goals has been a cross-cutting issue in the *International Technology Roadmap for Semiconductors* (ITRS). According to the Yield Enhancement Chapter of the 2003 ITRS,

*“The greatest challenge to a comprehensive data management system required for yield learning is the ability to deal with and integrate data streams that are continuous, periodic, sporadic, and interval based ...”<sup>1</sup>*

Software-based algorithmic techniques for merging data streams exist, but only provide limited certainties. More often, applications rely on time stamps to correlate data from different sources; consequently, an inaccurate time stamp renders the data invalid for many data mining and other analysis applications. One such application in APC is fault detection classification (FDC). To classify errors, the systems must be able to recognize a fingerprint for a specific type of fault, typically a set of causes in a specific order. The causes and effects may occur within mere milliseconds of each other, and FDC systems must be able to clearly distinguish the order of events for accurate diagnosis and correction to occur in real-time. Currently, significant variations in time stamp accuracies exist.

Diagnosing equipment faults will also require the cause-and-effect relationship to be accurately mapped. For e-Diagnostics, quality data complemented with accurate time stamps allow equipment engineers to sufficiently understand the scenario and process the tool remotely.

Automated material handling systems must also be synchronized with other factory information control systems. As chip manufacturers strive for near-zero idle time of equipment, materials should arrive at the equipment on time once a piece of equipment is ready to accept it. Not only must the equipment be able to accurately predict the time it will be ready to load the new material, other factory systems controlling the material must also be aware of the equipment status. Clocks among all interacting systems should be in synch with the equipment. This entails that the equipment clock remains “disciplined”; i.e., it not wander significantly from the time it predicted its ready state. The factory systems relevant to transferring the materials to the equipment on time must also have “disciplined” clocks to ensure that material arrives on time.

Factories must also maintain the typical network operations such as backing up data, securing the data, and accessing the factory systems, etc. Such applications require accurate time to be disseminated to the related software to execute the applications as efficiently and precisely as possible.

### 5.1 Chip Manufacturers Perspective

Cutting-edge chip manufacturing currently requires time synchronization on the order of seconds down to milliseconds. Interface A will become one of the critical drivers for accurate clock synchronization. The influx of data from 300 to 10,000 data points per second will require synchronization on the order of less than 1 ms. Extending factory automation for future applications will also require better coordination among different hardware and software tools. Improving automation will soon require a better shared sense of time within the factory. Maintaining factory efficiency will require tool usage data to be collected and correlated with tool failure data for rapid equipment fault diagnosis and repair. Ensuring minimal unscheduled

---

<sup>1</sup> Yield Enhancement Chapter, *2003 International Technology Roadmap for Semiconductors*, p. 14.

downtime of factory tools will also require material handling systems to readily provide materials by accurately coordinating with the tools. The coordination will require time synchronization between the manufacturing execution system (MES), scheduler/dispatcher, and the equipment. The clocks must be disciplined on all the systems to ensure that materials arrive on time and the equipment accurately estimates completion times.

While the requirements are evident, current equipment architectures and clocks lack the synchronization capabilities needed for efficient process manufacturing. With next-generation manufacturing requiring accuracy at the 1 ms level, chip manufacturers have expressed the need for better synchronized clocks *within* the tool. Current high-end factories with industrial Ethernet-based networks can achieve a maximum of 1 ms latency between any nodes on the factory floor. Having such networks enable the factories to fully exploit the features of NTP. The weakest link therefore remains within the equipment. Because accurate time is not accurately synchronized among the equipment operating system and subsystems, the equipment scheduler is thought to be the primary cause of lost and out-of-sequence messages.

Chip manufacturers often find time-stamped data from tools unreliable. Different systems or even subsystems are unable to coordinate in the most effective manner. Furthermore, time stamps are being generated at inconsistent points and not necessarily at the point where the data was measured.

### **5.2 Equipment Manufacturers Perspective**

With factory networks capable of supporting millisecond synchronization, equipment suppliers must ensure the clocks within the tools are able to receive the synchronization information from the appropriate time source. The synchronization accuracies achieved in tools are typically around 100 ms with significant variations, depending on the subsystem and CPU load. Current tools are generally not architected to support accurate clock synchronization. One of the key issues is the opaque nature of third-party hardware and software. Original equipment manufacturers (OEMs) rely on third-party suppliers who produce subsystems that are able to produce data, even time-stamped data, but no communication interface to synchronize the clock within the third-party system with the host clock. Significant latencies also tend to accrue between the host and subsystem. If the time stamps are created at the host instead of the subsystem, the time stamp is generally unreliable as latencies up to 2 minutes can occur before generated data reaches the host.

The overloaded, multi-tasking design of equipment architectures is another issue degrading the performance of clock synchronization within the tool. Multiple components, performing tasks simultaneously, are often designed to rely on the same CPU. Executing concurrent processes creates variations in CPU loading, and time stamp accuracy often degrades during a process upset. The irony lies in the error processing software. While one of the useful applications of accurate time stamps is for error debugging, error processing software introduces much of the non-deterministic CPU load. Often when synchronized time is most necessary, clock synchronization tends to be less frequent or non-existent. With the clocks running in an undisciplined manner, the time stamps become less reliable.

### **5.3 Manufacturing Software Perspective**

Next-generation factory automation software will become increasingly dependent on accurate time stamps. Future applications in FDC and e-Diagnostics will become especially reliant on accurate time stamps for process and equipment fault tracking and diagnosis. Extrapolating cause and effect will require data to be merged from heterogeneous sources and correctly ordered for accurate analysis. In addition, data generated from the tools are often used for data mining to achieve enhanced yield and quality of the manufactured chips. Effectively establishing the critical parameters and their respective tolerance values using data mining also requires quality data. Software developers confirm the need for time stamps to establish the order of events since data points are often buffered before reports are transported to preserve software performance.



Current developers agree the lack of reliable time stamps makes the software inefficient at best and ineffective at worst.

## **6 POTENTIAL SOLUTIONS**

The key to accurate time stamps is effective time synchronization and the ability to prove that a system is accurately synchronized. Clocks within the distributed system must maintain *accuracy* and *stability* over time. *Accuracy* is defined as the ability to conform to an ideal measured value. In terms of time, the ideal measured value is UTC. *Stability*, or precision of the clocks, indicates how well the oscillator can maintain the same frequency over a specified time interval. Stability is distinguished from accuracy because maintaining a consistent frequency does not necessarily mean the clock time is correct.

One method of improving temporal accuracy and stability is to ensure the availability of precision clocks on terminal devices and a protocol to reliably propagate, synchronize, and maintain accurate time throughout the network. The solution should be simple both to implement and to manage, as the other facets of semiconductor manufacturing are complex enough. The time stamp accuracy depends primarily on how applications are using the time stamp and how often a data point is being generated. Several methods exist for ensuring better timekeeping capabilities, and a more robust solution will most likely require a combination of accurate timekeeping mechanisms.

### **6.1 Time Synchronization Protocols for Distributed Systems**

A distributed system can be comprised of spatially separated electronic equipment or a single piece of equipment composed of multiple processors. Time synchronization among distributed systems requires a protocol to propagate accurate time and to maintain clock stability to all terminal nodes.

The goal of synchronization protocols is to distribute a common sense of time to all nodes within a system with a known worst-case accuracy. The nodes, or clients, can be PCs, workstations, servers, controllers, motion control systems, sensors, actuators, etc., that rely on correct time for synchronizing events or accurate time stamping. The common sense of time can be either relative or global depending on the application. The protocols must include utilities to determine the offset between a reliable “master” clock and “slave” clocks. Once the offset is determined, the protocol also includes algorithms to discipline the client clocks to achieve identical time information among the networked systems with as little disruption as possible. Networks with minimal variations in latencies yield the most accurate synchronization results in the distributed systems. Industrial networks relying on a non-deterministic Ethernet will need to take extra measures to ensure accurate synchronization.

### **6.2 Time Sources**

Distributed time synchronization protocols typically require an accurate time source and even diverse and redundant sources to ensure a sufficient level of fault-tolerance. *Reference clocks*, the clocks to which other network nodes synchronize, must be highly accurate and precise current time measurement devices. The reference clocks are typically GPS receivers or a cesium oscillator. 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 the National Institute of Standards and Technology (NIST) have the additional benefit of traceability to a national standard time or even international standard time. Accurate time sources can be acquired in various ways described below.

#### **6.2.1 Atomic Clock**

A resident commercial atomic clock provides the LAN with the best stability and is suitable for high-end relative synchronization requirements. Commercially available cesium atomic clocks

are accurate within a second for 30,000 years. The issue with resident atomic clocks is that they lack traceability to a national standards institute.

#### 6.2.2 GPS Receivers

GPS satellites provide traceable, 1  $\mu$ s UTC accuracy from an atomic clock to the client 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 clocks depend on U.S. operation of the GPS satellite constellation. The Galileo position system, a second GPS source, is currently being devised by the European Union and is expected to be fully operational in 2008.

#### 6.2.3 Radio Receivers

Radio broadcasts were used as a time source as early as 1904 by the U.S. Naval Observatory in Boston to aid navigation. 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. Accuracies range from 1 to 20 ms for HF receivers and 0.1 to 15 ms for LF receivers.

#### 6.2.4 Internet

Using NTP or NIST's 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*. Accuracy levels range between 100 ms and 2 s.

Any transmission through the Internet inherently has delays on the order of 50 ms to seconds because of the uncertainty of routing protocols. The Internet also requires an additional port to be opened for receiving the signal, which becomes an additional vulnerability if the rest of the company is protected by a firewall.

#### 6.2.5 Dial-Up

A dedicated dial-up line can provide an accurate time source within 15 ms accuracy. NIST provides the Automated Computer Time Service (ACTS) with a dial-up number available. Since time packets do not arrive in large files, the modem connection is more than sufficient for providing a source of traceable UTC time.

### 6.3 Precision Oscillators

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 CPU time needs to be dedicated to synchronizing the clock. Typically, quality clocks are inherently more stable and less affected by ambient factors and aging.

For computers, peripheral component interconnect (PCI) boards that are especially dedicated to providing accurate time are available. 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. The quality of the clock can be characterized by its accuracy and stability. The accuracy is limited by the precision or resolution of the clock and its ability to continuously follow the *Système International d'Unités* (SI<sup>2</sup>) second.

---

<sup>2</sup> SI is the modern metric system of measurement preferred in science and international commerce. The SI second is the base unit for measuring time and is defined as the duration of 9,192,631,770 periods of radiation corresponding to the transition between two energy levels of the cesium 133 atom.

Stability can be gauged by various methods. The *quality factor*,  $Q$ , of the oscillator is defined as its *resonance frequency* divided by its *resonance width*. Higher frequencies and smaller widths will result in a more stable clock. The *resonance frequency* describes the natural frequency of the oscillator. For example, the cesium element is 9,192,631,770 Hz; therefore, the current element serves as the standard atomic time. The frequency is based on the radiation cycles required for the transition between the energy levels based on the ground state of the cesium atom. The resonance width describes the range of possible frequencies that can trigger the resonator. Stability is measured based on the *Allan Deviation*; however, in layman's terms, it is easier to comprehend the time it takes for the clock to deviate by a set period of time, such as the highest deviation within one day.

### 6.3.1 Quartz Oscillators

The cost-effectiveness of quartz crystals makes them among the most widely used resonator in oscillators. Quartz clocks are infamously limited by aging, which affects long-term stability. Short-term stability is limited by environmental factors including temperature, humidity, pressure, and vibrations. Price and sufficiency for most applications make quartz clocks pervasive. General commercial quartz clocks in consumer products are often priced at less than \$1 and have a  $Q$  factor of  $1E+04$ , with compromised stability. The hardware clocks within PCs typically lose 5 to 15 s per day [3].

With an investment ranging from \$100 to \$2000, industrial quartz provides mechanisms to mitigate the ambient limitations. Oven-controlled crystal oscillator (OCXO) is a type of quartz oscillator designed to reduce ambient effects on the clock by enclosing the crystal in a temperature-controlled chamber. The stability of OCXO is limited by the noise from electronic components within the oscillator circuits. Another high stability quartz oscillator is the microcomputer-compensated crystal oscillator (MCXO) that uses a microchip to observe the frequency drift and to compensate for the drift. A less expensive version of the high stability oscillator is the temperature-compensated crystal oscillator (TCXO), which includes a temperature sensor to generate a correction voltage. The voltage generated induces a change in frequency to correct for the temperature effects. Synchronized to a UTC source, industrial quartz clocks have a  $Q$  factor around  $1E+06$ . The time it takes for an industrial quartz clock to be off by 1 ms ranges from 1 s to 10 days, depending on the type of industrial quartz.

### 6.3.2 Atomic Oscillators

Atomic oscillators are typically not cost-effective for terminal nodes, but may serve as a reliable source clock. Atomic oscillators are based on the resonance between two energy levels of an atom. An electromagnetic field is used to boost the atom from one energy level to the next. Conversely, an atom can emit energy to drop from its current level to a lower energy level.

Atomic oscillators have unparalleled stability and often deviate by about  $1E-06$  s in one day up to a month or more. The price, which ranges from \$8,000 to \$40,000, is a limiting factor for cost sensitive applications. Rubidium oscillators are the least expensive and most common commercial atomic oscillator. The resonance of the rubidium atom is used to control the frequency of a quartz oscillator. Rubidium oscillators are becoming smaller and less expensive, while providing better long-term stability than quartz-based oscillators. The  $Q$  factor of rubidium oscillators is around  $1E+07$ . The SI second is defined based on cesium oscillators, which have a  $Q$  factor of  $1E+08$ . The current NIST cesium atomic clock deviates no more than 1 s in 6 million years. Hydrogen maser oscillators have the highest  $Q$  at about  $1E+09$ , but have short term stability, less than the cesium clock.<sup>3</sup>

---

<sup>3</sup> Frequency reference masers, often chambers of ionized gas from a specific atom, are used in high precision frequency references such as atomic oscillators. Maser is an acronym for microwave amplification by stimulated emission of radiation. It is similar to a laser but functions at the microwave region, instead of the light region, of the electromagnetic spectrum.

Atomic clocks at the terminal nodes may soon be a reasonable component in electronic systems. NIST is currently developing miniscule atomic clocks for portable, highly precise timekeeping, which can be powered by an AA battery and will maintain accurate time to about 1E-07 seconds over the course of one day.<sup>4</sup> While the device will not be as accurate as its larger cesium clock predecessors, the portability and potentially lower manufacturing costs could make it more economically viable for a wider range of applications including efficient and error-free data transmission among wireless communication systems and any other distributed system with a potentially large number of nodes. The miniscule size of the atomic clock can be manufactured using existing techniques for assembling micro-electro-mechanical systems (MEMS). The final packaged clock spans only 1 cm<sup>3</sup> of space. Current commercially available atomic clocks are about the size of a cigarette box and used at cell phone base stations.

#### **6.4 Potential Solutions for the Semiconductor Industry**

Two major time synchronization protocols are available: NTP and IEEE 1588. The industry must also incorporate improved design of equipment architectures to ensure accurate time synchronization to the time-sensitive terminal nodes.

### **7 NETWORK TIME PROTOCOL (NTP)**

NTP is a time-tested and ubiquitous protocol for synchronizing distributed systems through the Internet. NTP supports virtually all platforms whether as part of the operating system (UNIX, MacOS) or 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 implementation. The Internet Engineering Task Force (IETF)—an open international community of network designers, operators, suppliers, and researchers—provides the technical details for implementing NTP. Detailed documents are published as Request for Comments (RFCs) with NTP version 3 (RFC 1305) and SNTP version 4 (RFC 2030). NTP version 4 implementations are available, but the RFC is currently a work-in-progress. Having evolved over 20 years, NTP has become a flexible network application capable of being configured and meeting widely divergent application demands. On a WAN, NTP provides time synchronization from 10 to 50 ms, nominally, while 1 ms accuracy can be achieved if a GPS is used as the time source within a 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.

#### **7.1 Architecture**

NTP is based on a scalable hierarchical client-server model using UTC as the source time. The *stratum* level, ranging from 0 to 16, defines how far a clock is away from a primary source. *Stratum 0* typically represents an atomic clock or a GPS receiving from an atomic clock. *Stratum 1* servers are the direct clients of stratum 0 and serve as the *time servers*.

Clocks residing on the same stratum can be *peers*, providing some increased fault tolerance through cross-checking and redundancy among clocks at the same level. Peers also synchronize among themselves based on the peer with the higher quality of time.

NTP distributes time through *unicast*, *multicast*, or *broadcast* messages over UDP/IP as long as their IP addresses are properly configured to receive the time signals. Clients can initiate a time request with a server and synchronize the local clock time over a series of time-stamped packets. Network administrators can also specify the time source to multicast or broadcast packets containing the current time at periodic moments. The default port assignment is 123 for NTP/SNTP.

---

<sup>4</sup> <http://www.boulder.nist.gov/timefreq/ofm/smallclock/>

## 7.2 Algorithm

### 7.2.1 Data Grooming Algorithms

NTP measures the time and transport latency delays on both the client and its respective servers. The *clock filter* algorithm determines the correctness of the client time stamp based on the minimal delay from a pool of measured roundtrip delays among the servers. The client clock is disciplined by an amortized adjustment over a period of time to prevent time warps. NTP also includes algorithms for selecting the optimal subset of time servers and combining the offsets to determine the clock adjustment. A quality metric and the variance relative to other servers are calculated for each server. Servers with outlying variances relative to other servers are eliminated, and new variances are calculated without the outlier until further discards no longer decrease the relative variances or the minimum number of servers remains.

### 7.2.2 Clock Discipline Algorithm

The NTP clock discipline algorithm is geared to minimize both the time and frequency offsets between the server and the client. The algorithm uses a feedback loop and measured offsets to adjust the local oscillator phase and frequency as close as possible to the time source.

NTP has fault tolerance capabilities built into the protocol. If several sources are available or if the designated source is unavailable, NTP will dynamically select the next best available source with which to synchronize. Clocks providing time stamps that are considered outliers will also be detected by NTP, and their time values will be discarded. Furthermore, if a network connection or a GPS time source is temporarily unavailable, NTP can still estimate the current time and offset of clocks based on history logs. NTP provides filtering techniques to mitigate latencies incurred among clients, servers, and their respective protocol stacks.

## 7.3 Authentication and Authorization

NTP offers authentication and authorization security options. When the administrator specifies a secure mode, each time synchronization packet has a 32-bit identification key and a 64- or 128-bit cryptographic checksum using the Data Encryption Standard (DES) or Message Digest (MD5) algorithms. The key and complementing checksums ensure only authorized systems are able to process the packets, thus preserving the integrity of the packets. NTP version 4 allows the use of *autokey*, which is a feature that uses random session keys and public-key cryptography.

## 7.4 Best Practices

Setting up NTP is relatively simple, and once it has been configured, management and maintenance are negligible. A reliable and accurate source to the local time servers is required. It is suggested that each subnet be configured to a time server. Redundant time sources, time servers, and paths from a node to a time server are also recommended. Unicast modes provide the most accurate and reliable service, while multicast and broadcast are typically sufficient for less stringent synchronization requirements. The advantage of multicast and broadcast modes is that the modes are more resource-efficient on the network and new systems do not require a custom NTP configuration file. Optimal polling intervals range from 1 to 20 minutes for local network paths. Modem paths are typically polled at intervals of thousands of minutes. Using SNTP at the leaf nodes is one way to minimize the synchronization traffic. In fact, SNTP should be used only at the network extremities.

Accuracies on the order of 1 ms within a LAN typically require a resident stratum 0 time source connected to NTP time servers on the factory floor. NTP primary servers must acquire time from a precision source such as radio or GPS receiver or modem. If SNTP servers are used, they should reside only at the root of a subnet with an integrated hardware time source. Other devices on another part of the network should not depend on the SNTP time server. While configuring a factory network for NTP can be intricate and challenging, monitoring and debugging tools are available to ease the protocol management. In addition, most problems are often detected, isolated, and repaired by NTP itself with its integrated management protocol.

## 8 SIMPLE NETWORK TIME PROTOCOL (SNTP)

SNTP is a lightweight subset of NTP, which is most appropriate for synchronizing leaf nodes with little variability in latency. SNTP provides a simplified access strategy to clients and servers configured with NTP by removing some of the filtering overhead. SNTP is designed to co-exist with NTP and be used within a dedicated system. With a radio clock integrated into the dedicated system, SNTP can achieve accuracies on the order of microseconds.

## 9 IEEE 1588

Initially developed by Agilent, IEEE 1588 (or precision time protocol [PTP]), has been designed for industrial Ethernet-based sensor/actuator LAN networks 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.

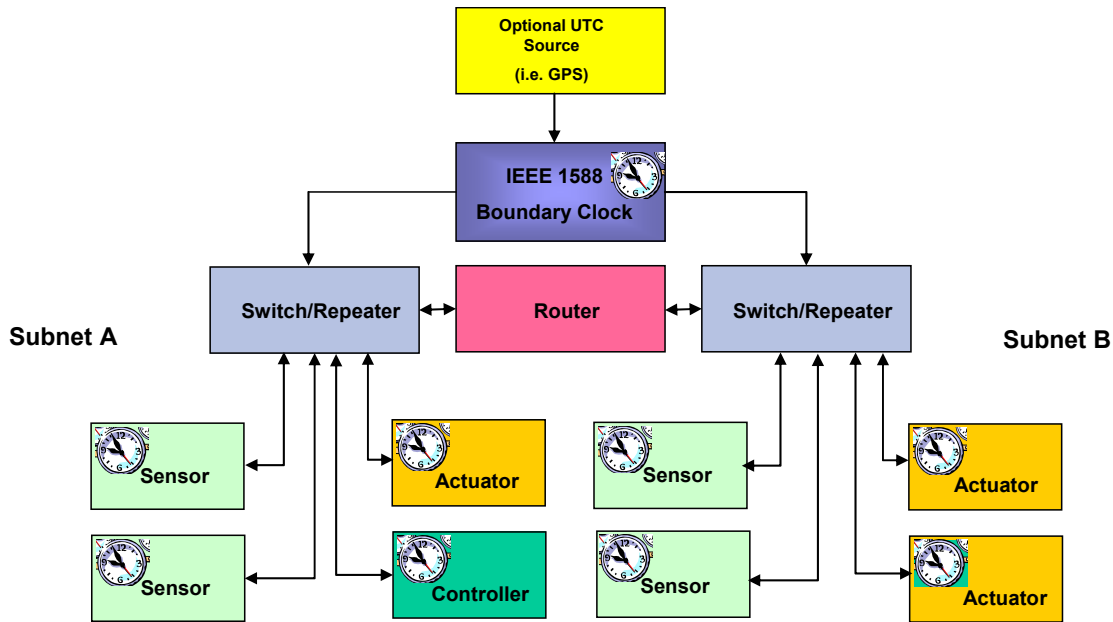
Manufacturing industries relying on precision motion control have traditionally implemented centralized command and control systems, which become costly to integrate and maintain due to diverse customer demands. As the automation equipment market has evolved to handle widely varied conditions and products, the modular approach has become the preferred architecture by manufacturers. The modular design allows end users to configure manufacturing systems with off-the-shelf technologies based on the functionalities required. Synchronizing modular subsystems from different suppliers has challenged many industries. Initially, proprietary solutions were devised to coordinate systems with microsecond accuracies, but such solutions have proven difficult to integrate because of the multitudes of demands and choices. The need has driven the development of a new standard time synchronization protocol, IEEE 1588, specifically for measurement and control systems.

Existing protocols such as NTP failed to support more stringent requirements or to readily integrate with industrial protocols. IEEE 1588 is a hybrid solution comprised of 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.

### 9.1 Architecture

IEEE 1588 has the master-slave architecture. The *grandmaster clock* is the most accurate clock available in the entire synchronization network at any time. *Master* clocks are designated as the most accurate clock in a subnet at a specific time. *Slave* clocks must be synchronized to the master clock on the same subnet. Because clocks are synchronized by subnets, sub-microsecond accuracies are achievable within a subnet. The standard specifies the implementation must allow the dynamic assignment of grandmaster and master clocks, which ensures a level of fault tolerance when a clock begins to fail. See Figure 2 for a simplified schematic.

IEEE 1588 is based on UDP/IP; in theory, it can be implemented on any bus system with multicast capabilities. Prototypes have typically been based on Ethernet. IP address implementation is not required at each node. Periodic multicast messages are sent from the master clock to its slave clocks for synchronization. Occasionally the slave clocks also propagate their time to the master clock to calculate the latencies.



**Figure 2** Simplified IEEE 1588 Architecture with Two Subnets.

### 9.1.1 Boundary Clock

Boundary clocks are dedicated devices for enhancing time synchronization by residing at the switch or router to mitigate the variable latencies of the typical network devices. The device eliminates significant network jitter, which can be up to 1 ms, and guarantees the synchronization packets are transmitted over physical devices with negligible jitter. Boundary clocks contain an on-board clock synchronized to the grandmaster clock. The grandmaster clock is fed into the *slave port* of the clock. The key feature of boundary clocks is to circumvent queuing time synchronization packets, which occurs in most generic network devices.

### 9.1.2 Node Hardware

IEEE 1588 also specifies hardware precision clocks and a time-stamping mechanism at the network card to minimize jitter generated by the protocol stack. Since network delays in industrial environments are typically more stable, they do not need to be calculated as frequently; thereby allowing the protocol to conserve computing and network resources. The hardware at the physical layer allows time stamps to be generated between the link and network layers of the protocol stack, thereby reducing millisecond fluctuations that can occur at the node. IEEE 1588 was created to support a wide range of clocks intended for high-end accuracies and for low-end, cost-sensitive applications.

## 9.2 Algorithm

The IEEE 1588 standard defines algorithms to ensure the following:

- *Low resource usage*
- *Low administration effort*
- *High synchronization accuracy within less than a microsecond*

### 9.2.1 Synchronization Algorithm

IEEE 1588 synchronization is comprised of two distinct parts to decouple the synchronization measurement and delay calculation tasks.

In the first part of the synchronization process, the master clock disciplines the slave clocks by sending a series of synchronization messages. Clock disciplining is a cyclical process in which synchronization messages containing a time stamp of when the messages are transmitted to a specific slave clock at defined intervals. The closer the time stamps are generated to the hardware the more accurate the estimate. The hardware portion of IEEE 1588 provides a method to eliminate the variations in the protocol stack by generating the time stamps close to the physical layer of the Open Systems Interconnection (OSI<sup>5</sup>) network stack. The master clock sends a *sync* message that contains an estimated time value of message transmission. The message detector, which ideally resides directly above the physical layer, time stamps the exact moment the message was received. A *follow-up* message containing the exact time of transmission of the corresponding *sync* message is sent to the slave. The slave calculates the offset based on the time stamp it generated at the exact time it received the *sync* message with both the time stamp value from the follow-up message and the one-way network delay subtracted. The slave clock is corrected over a series of *sync* and follow-up messages based on the calculated offsets.

The second part of the synchronization process measures the delay between the slave and the master. The slave transmits a delay request packet with the time stamp generated at the precise time of transmission to the master at irregular intervals. The master time-stamps the exact moment it receives the delay request message and calculates the slave-to-master delay by subtracting its time stamp from the delay request message. The master-to-slave delay is calculated based on the *sync* message. The one-way delay is based on the sum of the slave-to-master and the master-to-slave delay divided by two. The delay measurement does not need to be calculated as frequently as the offset measurements. The master and slave exchange a series of messages to accurately estimate the delay between the two nodes. The algorithm to determine the latency assumes a symmetrical delay.

#### 9.2.2 Best Master Clock (BMC) Algorithm

IEEE 1588 has also been designed to dynamically configure the topology based on the *BMC algorithm* to ensure low administration overhead and fault tolerance. The BMC algorithm establishes the master-slave relationships by comparing the characteristics of communicating clocks and setting the states of all local clock ports. The measured characteristics are based upon the clock's accuracy, layer, drift, and variance [4]. During synchronization messages, the current properties of the master clocks are transmitted to the slaves in the synchronization messages. States are calculated individually on each node so that the network can dynamically configure into a tree structure, with the most accurate clock, the grandmaster clock, at the root. The management portion of the protocol specifies the configuration and dissemination of the clock parameters. IEEE 1588 also supports redundant master clocks.

### 9.3 Best Practices

IEEE 1588 is designed for ease of management and has a set of best practices similar to NTP. Initial design of the topology is significant to the overall accuracy that can be achieved within the LAN. Hardware is recommended for sub-millisecond synchronization accuracy requirements. Maintaining small subnets and light network loads can also boost synchronization performance.

## 10 TIME SYNCHRONIZATION RECOMMENDATIONS

### 10.1 Time Source

Some considerations for selecting a time source include the precision, accuracy, and security of the time source. While the resolution and stability of the time source are also factors to be considered, well-known time sources typically have both to satisfy most applications. Since time synchronization protocols typically rely on symmetric delays for propagating time from a server to

---

<sup>5</sup> OSI architecture defines the reference model of network functions at each layer where a given protocol or protocols can be assigned.



a client, a symmetric delay from the originating time source to the time server residing on the LAN is even more important. Besides a resident atomic clock, the time source to offer the highest levels of accuracy and security, while ensuring the most symmetric delay of the two other choices, is GPS. GPS offers direct connection from the satellite to inside a company's LAN, which is often protected by a firewall, and provides UTC time. No intervening routers or WAN can cause unpredictable delays.

## **10.2 Network Synchronization Protocol Configuration**

Configuration of servers and clients play a significant role in the achievable synchronization accuracy of the overall factory system. Configurations with clients that can only listen degrade, 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. Therefore the clients should be able to propagate the local time to the server. The architecture is a critical factor in the overall accuracy achieved. The architecture must support the organization based on the size and topology of the network. Other network factors are support for redundancy, multiple subnets, large data volumes, and specialized time-sensitive applications.

The time sources and network paths to the time source must be diversified to achieve sufficient fault tolerance. Architectures with multiple time servers in a LAN offer more robust synchronization. Also if GPS signals are momentarily lost, it is useful to have time servers synchronize to one another or to have tiers of time servers based on their achievable accuracy for determining the actual current time. Time servers can also have integrated time references, such as a high quality atomic oscillator. The key is to ensure the time synchronization network is fault-tolerant when one source fails.

Selecting a time server also depends on the ability to simultaneously synchronize a number of clients during peak requests while still maintaining accuracy and availability. The time server must support the factory's industrial data protocol whether it is Ethernet, SECS/GEM, etc.

### *10.2.1 Enterprise Time Server Management*

Management of the time synchronization network requires the ability to monitor and control the server and clients and to provide a verifiable and secure audit trail. Network time servers should maintain a reliable level of determinism. The processing load of each server should be as balanced as possible. Enterprise security policies should be enforced for time just like any other network application. A central workstation provides the point of control, although management can be accessed through any web interface. The workstation should connect securely to the time source and to its clients. All the time servers and clients should 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)<sup>6</sup> as the unique identifier.

A verifiable audit trail assures that the network nodes were accurately synchronized at a specific point in time. This typically requires a dedicated audit trail server to ensure integrity of the clock synchronization and time stamping. The auditing server should be able to individually identify and monitor the nodes, record time, synchronization interval, accuracy offset, and the 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 provide immediate alerts when a monitored machine fails or swerves from desired tolerances.

## **10.3 Recommendations for Selecting a Protocol**

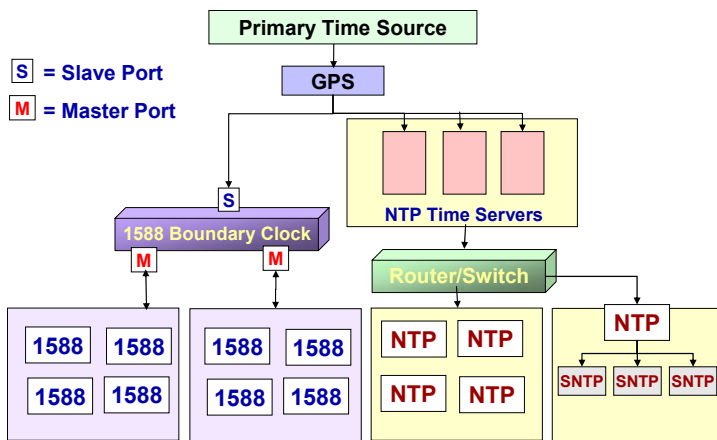
Heterogeneous systems often entail a range of different accuracy requirements. Often, the most cost-effective solution is to provide each system with the accuracy sufficient for a specific

---

<sup>6</sup> UUID is an Internet mechanism to uniquely identify and object or an entity. The UUID is comprised of a 128-bit number representing a reference to the network address of the host, a time stamp at which the identifier is generated, and a random element. The system is projected to generated unique identifiers until 3400 A.D.

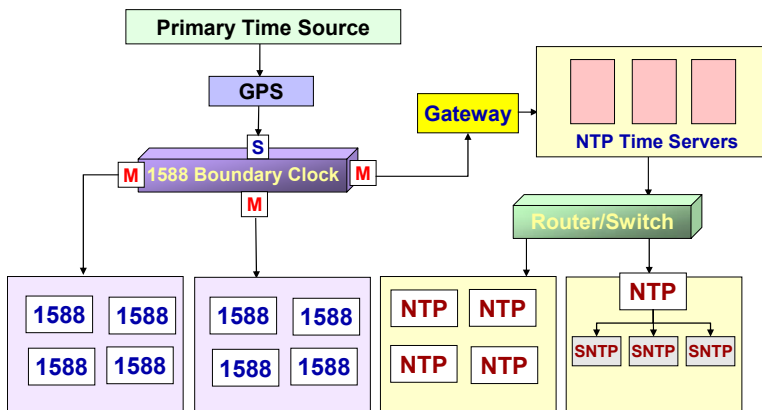
application. The protocol selection should be able to support a wide variety of accuracies such that costs can be cut on less time-sensitive applications, while ensuring the applications requiring the most stringent demands are met. Ideally, the protocol should be relatively simple to integrate with other synchronization protocols to preserve legacy systems, or it should be simple to use a sufficient, most cost-effective protocol for the application. Methods can be devised, such as a gateway mechanism, to integrate IEEE 1588 and NTP/SNTP. Such hybrid solutions are being demanded in the automotive industry, where accuracy on the order of 1 ms is required. One major car manufacturer has specified to their OEMs the need to be able to locally synchronize all clocks within a piece of equipment and to support SNTP, while the factory network is synchronized through IEEE 1588. Automotive OEMs can now design the new phase of equipment based on their customer demands. Chip manufacturers should also demand the synchronization protocol capability that best suits industry requirements.

Two protocols, such as IEEE 1588 and NTP/SNTP, can readily co-exist without much integration. Merely connecting the reference time source to an IEEE 1588 boundary clock and to the respective NTP servers would synchronize the systems. NTP and SNTP time information is transparent to each other. Figure 3 illustrates how the protocols can co-exist.



**Figure 3** Co-Existence of NTP/SNTP and IEEE 1588 in a Single Network.

Between IEEE 1588-enabled systems and NTP systems, a *gateway* that translates the packet information from one protocol to the other will be required. Figure 4 illustrates how a gateway can be used between a boundary clock and NTP time servers.



**Figure 4** IEEE 1588 as the Primary Protocol.

When selecting a protocol, some of the critical elements to look for, all of which are available in NTP and IEEE 1588, are as follows:

- *The algorithm must have access to a time base.* Depending on the network, the time base can be either *global (or external)* or *internal*. A global time base is typically described as UTC time. Global time is a significant requirement for systems distributed among different locales. For industrial purposes, where the systems are typically confined to a LAN, internal time synchronization is sufficient. However, even in internal synchronization, a UTC time base provides the most stable time source.
- *The algorithm must be fault-tolerant, having the capability to cope with faulty clocks and unreliable networks.* Especially important in real-time clock synchronization, a sufficient degree of fault tolerance can be realized through various software and hardware mechanisms. The three selected protocols all use delay approximation to determine the network delay between signal birth and termination. This is necessary for the correct time to be estimated; if a time stamp signal takes 1 microsecond to arrive at its destination, the synchronization of time between the two components is already off by that 1 microsecond. Because the correction to offset delay is calculated by assuming that the signal's roundtrip takes equal time per segment, a lopsided network (where signals take longer to either embark or arrive) will cause more errors than a balanced network. Variations in latencies in the network create a network imbalance. The protocols must be able to evaluate the consistency of the synchronization clocks. If outliers or unresponsive servers and nodes are detected, the algorithm renders the clocks invalid. Furthermore, clocks must attain credibility over time based on their stability. Reliability of a measured time is typically established over a series of packet exchanges until the server's replies satisfy a specific set of conditions enumerated in the protocol.
- *The algorithm must allow for small measurements of time even during the synchronization to avoid time warps.* During time synchronization, the clock must provide the accurate time without interruption. When a time discrepancy between a source clock and system clock is detected, the system clock should not fully adjust itself by the offset; this might result in time warps where time stamp gaps and duplication can occur. For example, if the clock were running ahead of the master clock by 1 second when synchronized, then the hard backward adjustment to the clock could cause two events to have the exact same time stamp: one before the adjustment and one a second after the adjustment. The process for clocks to rectify their time perception is often to adjust the time increment through continuous amortization. The time increment essentially determines how fast the software clock runs; this will be unique for every hardware clock and is configured by the algorithm through a series of setup synchronizations. When this increment is changed, instead of adjusting the time on the clock, this variable gradually slows or speeds up the clock to smoothly adjust the registered time to that of the master clock. This ensures a continuous flow of apparent time without any hard interrupts or time warps.
- *The algorithm must be efficient so that it does not degrade system performance.* Packet sizes, time synchronization analysis, and synchronization frequencies should be optimized to maintain network performance. Typical synchronization protocols, such as NTP, have been measured to use less than 0.41% of a T1 network bandwidth and 1.54% of the available CPU time [5]. Time synchronization protocols typically exist over UDP/IP to ensure efficient use of network resources. While TCP/IP provides guaranteed transmission of packets from source to destination, the retransmission of packets can become overwhelming to loaded networks especially when time synchronization occurs frequently. Another feature enhancing the efficiency of the protocol is multicasting and broadcasting. Most protocols support both features, with multicasting being the most

efficient method for networks since it selectively sends synchronization packets to interested end nodes as opposed to sending it to all nodes whether they require synchronization or not.

#### **10.4 Network Topology Recommendations**

Designing the topology for synchronization will require an understanding of traffic levels, patterns, and performance on the factory floor. Understanding congested areas will be especially useful in strategically placing IEEE 1588 boundary clocks or NTP/SNTP servers for optimal cost-efficiency.

#### **10.5 Equipment topology recommendations**

To properly synchronize time, the equipment should have the core capabilities to support factory-wide system synchronization. The equipment should be able to locally synchronize all clocks within the tool. Depending on the complexity of the equipment and the accuracy required, a lightweight protocol is typically sufficient. One method to consider is the use of a sensor bus to propagate time information, such as SNTP or IEEE 1588 packets, between the sensors and actuators. The equipment should be able to be configured to locate a synchronization source both dynamically and manually. After a time source is designated, the equipment must be able to accept this time information and be programmed to poll for the time both periodically and when an error is suspected. In addition, the equipment must be able to propagate its network address to its designated network time source. The equipment should rely on dedicated hardware clocks while the equipment is running, *not* on an error-prone pure software clock. Selecting a cost-effective, precision clock with accuracy and stability based on the SI second is another critical factor in ensuring successful coordination and accurate time stamps among distributed systems.

Operating systems within the equipment should also ensure time synchronization is not easily compromised by overloading system tasks. Real-time operating systems provide better timing guarantees and are useful for time-critical applications.

### **11 TIME STAMPING**

#### **11.1 Standard Time**

Sharing information among heterogeneous systems spanning distributed sites requires a single reference, often Coordinated Universal Time or Temps Universel Cordonne (UTC). UTC provides the common denominator for dispersed systems that must cooperate towards a specific goal.

As the predominant standard timekeeping system, Greenwich Mean Time (GMT), which represents the time at the Earth's prime meridian or zero degrees longitude based on the earth's rotation, has become obsolete because of the minute idiosyncrasies in earth's movement. UTC, effective January 1, 1972, introduced the "leap second" in an attempt to couple the earth's rotation with atomic clocks. When the difference between the atomic time and the time based on Earth's rotation approaches 1 second, which usually occurs about every 18 months, a leap second is adjusted in UTC, thus maintaining synchronization between the highly stable time system and the Earth's rotation.

UTC is a *universal* time, meaning that it has the same value anywhere in the world. To extrapolate the local time, one has to add or subtract a constant based on the local time zone. The time value for UTC is determined based on a *coordination* of input from several national standards institutions around the globe. More than 30 standards institutions provide their time values from 200 national atomic clocks and submit them to Bureau International des Poids et Mesures (BIPM) in Sevres, France, to create the International Atomic Time (TAI). TAI relies on the continuous, as opposed to discrete, counting of the SI second. The TAI is then 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, which is currently 32. For applications in which relative synchronization is

more important than global synchronization, it might be useful to consider TAI as the preferred time scale, since the leap seconds inherent in UTC can create coordination problems.

### 11.2 Time Representation

Many nations have different methods of representing 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 8601:2000, *Data Elements and Interchange Formats*, specifies a standard framework for numeric representation of date and time to avoid confusion in international communication especially for increasing portability of computer communication applications. Time representation according to ISO 8601 has already been adopted internationally. The standard does not require compliance to a specific format, but rather defines a framework for representing date and time.

Some current common practices include the following:

- **Date:** *YYYYMMDD* where *YYYY* represents the 4-digit year based on the Gregorian calendar, *MM* is the month and can have a value from 01 to 12, and the *DD* is the day from 01 to 31.
- **Time:** *hhmmss* where *hh* represents the hour from 00–24, *mm* represents minutes from 00–59, and *ss* represents either *integer or floating point* seconds from 00–60. The representation for midnight is typically 00 as opposed to 24, but the option exists for either convention. The 60 accommodates leap seconds, which are designated every few years by the IERS. For applications requiring sub-second accuracies, any fraction of a second can be represented by placing a decimal after the two-digit representation of a whole second. Therefore, the time stamp 000000.001 represents 1 ms after midnight. Specifying a lowest resolution for time stamps might be useful to ensure a standard level of data quality for the applications residing in the distributed system. Ideally, the time stamp should have at least the same resolution as the most accurate clock in the network to fully exploit the accuracy capabilities available.
- **Date and Time:** *YYYYMMDDThhmmss* allows the date and time to be represented in a single field. The date and time can be separated by a “T.”

### 11.3 Epoch

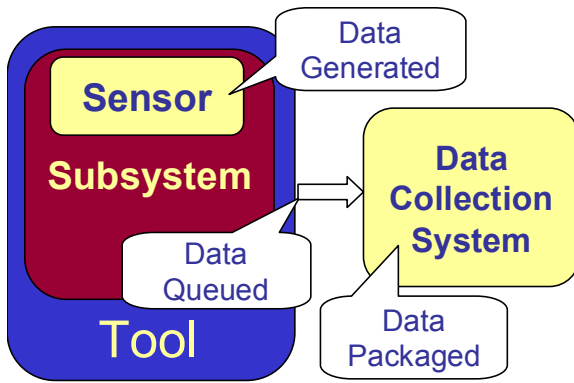
The epoch is the reference date based on an event for measuring the passing of time. Some common epochs used are the NTP epoch, which began on January 1, 1900, while the GPS epoch began January 6, 1980. A consistent epoch among each system ensures the time stamps are consistent and eliminates further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

### 11.4 Time Stamping

Timestamps coupled with a piece of data should reflect the time the data was generated in order to provide a sufficient level of data quality. Time stamps should not include any of the following:

- a) Time a piece of data is in a transmission or processing queue
- b) Transmission time
- c) Data processing time

See Figure 5 for some time-stamping points.



**Figure 5** Possible Time Stamping Points.

## 12 POTENTIAL STANDARDS EFFORTS

Meeting time synchronization demands in an environment comprised of multitudes of different hardware and software from various suppliers will require guidelines and standards to efficiently produce a common sense of time.

Before or with standards efforts, clock synchronization and time stamping usage scenarios must be identified and prioritized to clearly understand the accuracy and precision requirements. Cost-benefits analysis should also be conducted to determine if standards solutions and guidelines are feasible. It is also necessary to ensure current Semiconductor Equipment Materials International (SEMI) standards will not conflict with other standards efforts and to leverage existing SEMI standards for the development of time synchronization and time stamping standards.

Standards efforts geared towards improving equipment clock synchronization can further improve the synchronization of the overall networks and ensure accurate time stamps. Developing a standard interface and format within SEMI's Sensor Bus would help to communicate time information to components within the tool. Specifying a data exchange format for uniquely identifying a clock and its characteristics such as accuracy, precision, and worst-case drift would also prove advantageous. IEEE 1588 already has a standard method of uniquely identifying a clock and its features. Any new format representing a clock model that is developed in the semiconductor industry should use the standard as a basis. A guideline for architecting equipment for better clock synchronization also expedites the development of equipment architecture to facilitate factory time synchronization and time stamping.

As time stamps must be communicated among heterogeneous hardware and software tools, standard procedures and formats would promote the efficient communication of time information. Defining a time and date format based on industry needs and common current practices would prevent the need to translate between various formats and would ensure that the integrity of the time stamp is preserved. ISO 8601 helps define a framework, but it is up to the industry to define the exact format and epoch suitable for equipment-based communications. Ensuring data quality may also require a method to convey time stamp quality. Time stamp traceability requirements should also be specified to certify synchronization performance is accurate to NIST's or the relevant national standards institute's atomic time. The format should also encapsulate at what point the time stamp was attributed to a data point. The quality of the time stamp should be judged based on the accuracy, precision, and traceability of the clock as well as the proximity of the time stamp to the point the data was generated.

## **13 CONCLUSIONS**

Devising an effective time synchronization scheme within the factory and developing time synchronization and time-stamping standards should be a multi-disciplinary effort. Input from factory process engineers, factory network administrators, OEMs, software developers, and other factory automation stakeholders must be assessed to fully define the requirements and to ensure each need is met in the most resource-efficient manner.

### **13.1 Accuracy and Precision Requirements**

Time stamp accuracy requirements depend on the data rate, precision, and resolution of all related systems, where the data would potentially need to be merged. The data collection rate should help to define time stamp accuracy, precision, and resolution with a fine enough level of granularity so that collisions will not occur when the data streams are merged based on time. The granularity should be at least an order of magnitude greater than the data rate. Accuracy requirements can be determined by assessing the types of analysis applied to the data to see the frequency needed for data collection. Making accuracy requirement decisions also necessitates the understanding of system accuracy capabilities. For instance, the resolution of the clock or time stamp must be known, since it could limit the best accuracy achievable unless the system capabilities are modified.

### **13.2 Cost**

While the costs are relatively low to maintain millisecond accuracies within a network, ensuring the highest synchronization accuracies overall to get useful time stamps can incur significant costs. Tools and networks may have to be re-architected to sufficiently exploit the capabilities of the time synchronization protocol. Hardware enhancements for each component requiring high-end precision time may be required to achieve microsecond to sub-microsecond accuracies.

### **13.3 Next Steps**

Synchronizing clocks among factory systems requires a robust, accurate, and precise reference time source and a method for disseminating the reference time to all the network nodes. The GPS receiver serves as the most secure and accurate reference time source. However, when GPS signals cannot be received, it is also advisable to have redundant and diverse sources and paths to the sources if a time signal is lost or the receiving equipment fails.

The method for propagating an accurate reference time throughout the network can be comprised of IEEE 1588 for next-generation applications requiring high-precision and sub-millisecond accuracies, NTP for applications with a worst-case accuracy requirement of 1 ms, and SNTP for dedicated systems at the extremities of the network that do not need to be synchronized with systems outside of its subnet. The factory network and equipment should strive to support IEEE 1588 and NTP/SNTP. Commercially available IEEE 1588 products are projected to arrive in early 2005, and improvements to the standard based on user requirements are also helping the standard to evolve. The IEEE 1588 User Requirements Task Force has been examining how the standard is meeting the timing needs of various industries and is open to making the standard a better fit for the semiconductor industry. The semiconductor industry should actively participate in the IEEE 1588 standard to understand the current synchronization capabilities and to ensure that industry needs are met.

The most critical issue that needs to be addressed lies within the equipment. Better quality precision oscillators inside the equipment will require less frequent synchronization and therefore will save the bandwidth and computing resources for other process-critical applications. Each equipment clock should be able to characterize itself based on its precision, resolution, worst-case accuracy, and latest time of synchronization. The equipment clock should be able to report such information on demand to judge the time stamp quality of a specific piece of data. The communication architecture of the tool controllers, subsystems, actuators, and sensors must be able to receive and propagate time information to the host controller. The equipment controller in turn must be able to propagate and receive time information to and from the network. Current

methods for passing messages among the tool components, should therefore accommodate the time synchronization packets.

Equipment systems generating data will also require the capability to time stamp the piece of data as close as possible to the actual time it was created. Standardizing the equipment data time stamp format and reference epoch will also reduce data processing time and eliminate translation ambiguities, greatly easing the data merging process. Chip makers should actively demand time synchronization capabilities in their tools to maximize their automation capabilities. It is imperative for the equipment suppliers that future equipment architectures be designed to be more amenable to precise and accurate clock synchronization. In the short-term, suppliers should be prepared to communicate their time stamp precision and accuracy to meet chip maker demands.

Aside from equipment issues, the components to a robust and efficient time synchronization solution are readily available and evolving to meet ever increasing demands from a diverse array of industries. With NTP being widely deployed over the Internet and enterprise networks and IEEE 1588 on the horizon, it is a timely opportunity for the semiconductor industry to leverage these solutions to enable efficient, more intelligent, and less error-prone automation for realizing future cutting-edge chip manufacturing.

#### 14 REFERENCES

- [1] Li, Q. and Rus, D. "Global Clock Synchronization in Sensor Networks." *IEEE Infocom*, 2004.
- [2] Eidson, J.C. and Hamilton, B. "IEEE-1588 Node Synchronization Improvement by High Stability Oscillators." *Workshop on IEEE-1588, Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, September 24, 2003, pp.102 – 112.
- [3] Lombardi, M. "Computer Time Synchronization," NIST web page, 1999.  
<http://www.boulder.nist.gov/timefreq/service/pdf/computertime.pdf>
- [4] Mohl, D. "IEEE 1588: Running Real-time on Ethernet."  
<http://ethernet.industrial-networking.com/articles/i17real-time.asp>
- [5] Mills, D. "NTP Performance Analysis."  
<http://www.eecis.udel.edu/~mills/database/brief/perf/perf.pdf>
- [6] Bregni, S., *Synchronization of Digital Telecommunications Networks*. John Wiley & Sons Ltd., England, 2002, pp.2, 3.
- [7] *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, IEEE 1588-2002 Standard, November 2002.
- [8] "IEEE 1588-2002 Standard for A Precision Clock Synchronization Protocol for Networked Measurement and Control Systems," June 2004. <http://ieee1588.nist.gov>
- [9] "Information on ISO 8601, the date and time representation standard,"  
<http://www.cs.tut.fi/~jkorpela/iso8601.html>
- [10] "Introduction to Quartz Frequency Standards - Oscillator Comparison and Selection."  
<http://www.ieee-uffc.org/freqcontrol/quartz/vig/vigcomp.htm>



- [11] Kuhn, M. "A Summary of the International Standard Date and Time Notation."  
<http://www.cl.cam.ac.uk/~mgk25/iso-time.html>
- [12] Lee, K. and Eidson, J. *Workshop on IEEE-1588, Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, NISTIR 7070, September 24, 2003.
- [13] Lombardi, M. *NIST Time and Frequency Services*, NIST Special Publication 432, 2002.  
<http://www.boulder.nist.gov/timefreq/general/pdf/1383.pdf>
- [14] *Network Time Protocol Version 3*, RFC 1305.
- [15] NIST Time and Frequency Division, July 2004.  
<http://www.boulder.nist.gov/timefreq/index.html>
- [16] SEMI standard E54 - Sensor Actuator Network Standard,  
<http://wps2a.semi.org/wps/portal>.
- [17] *Simple Network Time Protocol Version 4*, RFC 2030.
- [18] Spada, S. and Shah, H. "IEEE 1588 Lowers Integration Costs in Continuous Flow Automated Production Lines," *ARC Insights*, August 20, 2003.
- [19] Strand, B.C. and Fralic, G. "Economic Benefits from Advanced Quality Control of TMP Mills," [http://www.pacsim.com/Articles/TMP\\_AQC/TMPAQC\\_article.shtml](http://www.pacsim.com/Articles/TMP_AQC/TMPAQC_article.shtml).
- [20] Strand, B.C. "Analysis of Pulp and Paper Quality Data," *TAPPI Paper Making Conference*, San Francisco, 1994. <http://www.pacsim.com/Articles/Quality/Quality.HTM>
- [21] Symmetricom, Inc. – Timing, Test and Measurement Division. "Network Time Synchronization: 5 Essential Elements," *Information Week*, January 1, 2004.  
[http://whitepapers.informationweek.com/detail/RES/1082053388\\_675.html](http://whitepapers.informationweek.com/detail/RES/1082053388_675.html),
- [22] The Network Time Protocol (NTP) Distribution, March 2004.  
<http://www.eecis.udel.edu/~mills/ntp/html/index.html>
- [23] "Time Synchronization in Switched Ethernet." *The Online Industrial Ethernet Website*,  
<http://ethernet.industrial-networking.com/articles/i07switched.asp>.
- [24] "What time is it?" *Bureau International des Poids et Mesures Website*.  
[http://www.bipm.fr/en/scientific/tai/time\\_server.html](http://www.bipm.fr/en/scientific/tai/time_server.html)