SynUTC and IEEE 1588 Clock Synchronization for Industrial Ethernet

Roland Höller
Institute of Computer
Technology, Vienna
University of Technology
Viktor-Kaplan Str. 2, 2700
Wiener Neustadt, Austria
roland.hoeller@tuwien.ac.at

Thilo Sauter
Institute of Computer
Technolgy, Vienna
University of Technology
Gußhausstraße 27-29,
1040 Vienna, Austria
sauter@ict.tuwien.ac.at

Nikolaus Kerö
Oregano Systems
Design & Consulting
GesmbH
Phorusgasse 8,
1040 Vienna, Austria
keroe@oregano.at

Abstract

This article deals with two mechanisms for time synchronization over Ethernet networks, the "IEEE 1588 standard for a precision clock synchronization protocol for networked measurement and control systems" and the "synchronized universal time coordinated" technology. It presents a prototype system, which supports both synchronization mechanisms by offering dedicated hardware support on Ethernet network nodes and Ethernet Switches respectively. Finally the two technologies will be compared with respect to some key properties.

1. Introduction

Temporal relationships have always played an important role in industrial control and automation systems. After all they have been a key issue in the development of fieldbus systems. In the resent past, however, Ethernet technology has become popular also in the field level [10,11,12,13], offering a cost effective, yet high-speed technique for interconnecting sensors, actuators, and computers distributed in space together with the possibility of achieving a vertical integration of the field level components into standard office level networks and applications. As systems become more and more complex with sensors, actuators, and computers wider distributed in space and communicating via a network, robust operation as well as simultaneous triggering of events and synchronous data acquisition at several nodes still require tight clock synchronization between all related network nodes.

On one hand the SynUTC (Synchronized Universal Time Coordinated) technology enables fault tolerant high accuracy distribution of GPS time and time synchronization of network nodes connected via standard Ethernet LANs [1,4,5,6,7]. By means of exchanging data packets in conjunction with hardware support at network nodes and network switches, an overall worst-case accuracy in the range of some 100 ns can be achieved, with negligible communication overhead. This technology thus improves the 1 ms-range accuracy achievable by conventional, software-based approaches like NTP by 4 orders of magnitude. Applications can use the high-accuracy global time provided by SynUTC for event timestamping and event generation, both at hardware and software level [2,3,6,7].

Furthermore the IEEE standard 1588 has recently been approved, which specifies a dedicated protocol, the Precision Time Protocol (PTP), for Ethernet networks to synchronize clocks in networked measurement and control systems [9].

Both the IEEE 1588 standard and the SynUTC technology are based upon taking highly accurate time information into account, when dedicated data packets pass at the media independent interface (MII) between the physical layer transceiver and the network controller upon packet transmission and reception, respectively [6,7,8,9].

2. SynUTC Technology

The SynUTC technology for high-accuracy clock synchronization is based upon a few key paradigms, which will be briefly introduced subsequently [1,2,4,6].

2.1. Adder-based clock

Local time at any computing node is maintained by means of an unconventional adder-based clock, which uses a high-resolution adder instead of a simple counter for summing up the elapsed time between succeeding oscillator ticks. Owing to this, the clock can be paced by an oscillator with arbitrary frequency. Moreover, the local clock is fine-grained rate adjustable in steps of nsec/sec and supports state adjustment via continuous amortization.

2.2. On-the-fly timestamping

Clock synchronization over packet-oriented networks requires timestamping of data packets (Clock Synchronization Packets, CSPs) used for time transfer. In purely software-based clock synchronization, timestamping at the sending respectively receiving side is done by reading the clock when assembling the CSP for transmission respectively in the packet reception interrupt service routine. This implies that the transmission delay uncertainty ε includes both the network channel access uncertainty and the reception interrupt latency, which can be quite large. As a consequence, the achievable synchronization precision is quite poor (ms-range).

This approach performs timestamping on-the-fly, when a CSP is sent respectively received by the network controller. As the Ethernet frame is received nibble by nibble via the MII interface, the special hardware scans the incoming data for the start frame delimiter (SFD) of the packet. With the clock cycle it encounters the SFD, the adder-based clock is triggered to store its current value in a 96 Bit timestamp register. This data is inserted in the "Receive TS" field of the CSP during reception or in the "Send TS" field of a CSP during transmission.

2.3. Interval-based paradigm

A unique feature of this approach is the support of the interval-based paradigm [8]. Real-time t, that is, GPS time or UTC, is not just represented by a single time-dependent clock value *C(t)* here, but rather by an accuracy interval ***C(t)*** that must satisfy *t* in ***C(t)***. Interval-based clock synchronization thus assumes that each node *p* is equipped with a local interval clock $\mathbf{C}_p$ that continuously displays *p*'s instantaneous accuracy interval $\mathbf{C}_p(t) = [C_p(t) - \alpha_p^-(t), C_p(t) + \alpha_p^+(t)]$. Naturally, $C_p(t)$ is just the local clock value and $\alpha_p(t)=[-\alpha_p^-(t), \alpha_p^+(t)]$ the negative and positive accuracy maintained by two additional adder-based clocks.

2.4. Clock state and rate synchronization

An interval-based (external) clock synchronization algorithm is in charge of maintaining any node's local clock such that a worst case precision and accuracy is guaranteed. Moreover, we synchronize not only the state of the clocks in the system, but also their progression: An additional interval based clock rate synchronization algorithm is employed, which achieves high synchronization precision without expensive OCXOs (Oven Controlled Crystal Oscillators) as the nodes' frequency sources and frequent resynchronizations.

3. IEEE 1588 PTP

The recently approved IEEE 1588 standard specifies a method to synchronize a number of real time clocks, which are interconnected via a packet oriented network [9]. PTP is targeted to Ethernet, but can be applied to any other packet oriented protocol that supports multicast addresses. The PTP is optimized for:

- small networks consisting of a small number of subnets
- consuming a small amount of resources with respect to computing power, bandwidth, and memory in the network nodes
- minimal administrative overhead

The mechanism has been developed within the test and measurement division at Hewlett-Packard, and the basic idea was to allow interconnected measurement devices, with synchronized clocks, to sample data and assign a timestamp to it. The IEEE 1588 standard basically consists of five parts:

- a means to automatically partition a PTP network
- definition of the PTP clock
- synchronization of the PTP clocks
- best master clock algorithm (BMC) to define the master clock
- PTP network management protocol

A core element of the PTP lies in the ability to segment a network by setting certain PTP clocks in a passive mode, to allow the calculation of point-to-point delays.

The standard does not distinguish between a clock, which is realized in hardware or in software. For high accuracy clock synchronization in the sub-microsecond range, however, dedicated hardware support similar to the SynUTC clock is imperative and proposed for Ethernet networks.

Synchronization is based on a master-slave principle. Among all connected and accessible nodes one master clock is selected using the best master clock algorithm. The master starts sending synchronization messages on a regular basis. The synchronization message contains the time, when the message left the master node (a more accurate time, which excludes all inaccurate protocol stack latencies, can be sent in a follow up message). The receiver now is able to calculate the correction values for his local clock by taking the timestamp of the sync message, the timestamp of the follow-up message, and his own clock into account.

4. System Overview

The prototype system consists of four network nodes, a COTS Ethernet switch, a time synchronization add-on to the switch, and a GPS-receiver (see Figure 1).

The system nodes synchronize time via sending Clock Synchronization Packets over the Ethernet LAN. The clock synchronization software is executed on the 32-bit Altera NIOS CPU. Besides executing the synchronization algorithm itself, the software must also measure system parameters like the transmission delays and the clock drifts. The SynUTC synchronization algorithm is round-based; one round comprises the following activities: [6]

- The full message exchange, where every node broadcasts a clock synchronization packet with its own clock value and accuracy interval to its peers.
- A message collection phase where the node waits for the messages of its peers.

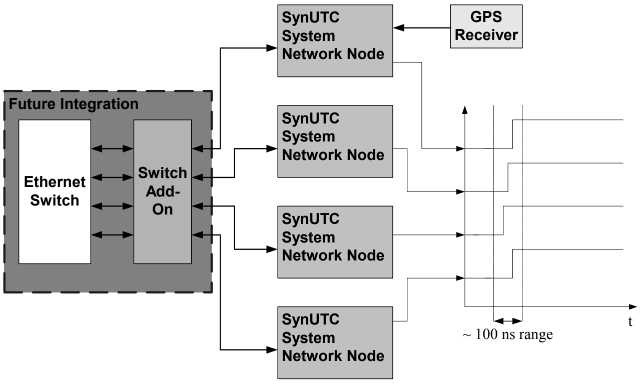
SynUTC System Network Node
GPS Receiver
Future Integration
Ethernet Switch
Switch Add-On
SynUTC System Network Node
SynUTC System Network Node
SynUTC System Network Node
~ 100 ns range
t

Figure 1. Overview of the clock synchronization prototype system.

- The computation of the new clock value and accuracy interval by a fault tolerant interval intersection function.
- The re-synchronization itself where the computed clock value is taken over.
- The free-run phase in which the clock simply runs without any interference from the algorithm.

When CSPs are sent out from the network interfaces of the nodes, they arrive at the input ports of the prototype system's switch add-on. There a timestamp is inserted in the Ethernet packet on the fly. When the packet is forwarded to its destination, the amount of time the packet spent on the Ethernet switch is determined by the switch add-on and this information is inserted in the data field of the packet.

As the building blocks of the prototype system are completely compatible to IEEE 802.3 the PTP protocol stack can run without any limitations on the network nodes. Furthermore the adder based clocks in the ASICs of the nodes are compatible to the IEEE 1588 time format, which uses a 64-bit time representation, where the upper 32-bit part is the number of seconds and the lower 32-bit part is the number of nanoseconds.

5. Network nodes

Typically the network node is intended to be used to connect to e.g. a sensor or an actuator in an industrial automation network, where space and power requirements have to be taken into account. To be able to develop a flexible prototype system node that contains also the CPU on the chip, reconfigurable hardware is used as the centerpiece of the network interface card. This processor is an Altera NIOS 32-bit microcontroller core, which runs uCLinux as operating system.

Implementing a clock synchronization service according to the ideas outlined above, requires support on every node, which can be provided in a single System on Chip (SoC).

It hosts the following functionality:

- A high-resolution adder-based local clock with a mechanism for linear continuous amortization.
- Two additional adder-based clocks holding and automatically deteriorating the bounds on accuracy with respect to the external reference time.
- An interface to a GPS timing receiver, made up of a 1-pps digital input and a RS232 serial interface.
- Application-level event generation and timestamping capabilities.
- A standardized interface for packet timestamping near the physical layer (IEEE 802.3 MII) [8].
- An Ethernet MAC Controller.
- An on-chip CPU running the synchronization algorithm and protocol stacks according to IEEE 1588.

To make full usage of the advantages of the SynUTC technology, but at the same time allow other IEEE 1588 compatible devices to join the ensemble for time synchronization, implementation details of both methods have to be taken into account. E.g. IEEE 1588 PTP Management messages have to be used additionally for communication and synchronization, or the adder-based local clock has to use a certain time representation.

The adder based local clock is basically a 96-bit adder. It has several modes of operation to support the clock synchronization algorithm. The synchronization accuracy can be optimized by increasing the clock frequency that drives the adder based clock core in the network node. Other than simply incrementing the contents of the register by one with every rising clock edge, a high resolution clock register is incremented by a programmable register value. It can be modified to any value for rate adjustment purposes, with speeding up and slowing down the clock.

To increase the clock frequency of any integrated digital circuitry the number of logic levels between two storage elements (Flip-Flops) has to be minimized in order to reduce propagation delay through the logic. A commonly used method to accomplish this is to use pipeline stages to split the amount of logic that has to be passed in one clock cycle.

With this it is possible to achieve acceptable frequencies, even in FPGA (Field Programmable Gate Array) devices, which are approximately a factor of 6-8 slower than comparable ASIC technologies. Our evaluation of several technologies revealed that pipelining has definitely to be used to achieve synchronization within the 100 ns range.

Furthermore events occurring at dedicated input pins of the network interface card chip can be timestamped. Conversely dedicated output pins can be activated at programmable points in time.

6. Synchronization Switch

As already depicted in Figure 1, the first prototype system will operate with a common of the shelf Ethernet switch and an add-on for delay measurement. To allow the implementation of IEEE 1588 boundary clocks the realization of a special Ethernet switch is imperative and will be explained here.

When network nodes are interconnected in standard Ethernet networks, switches are commonly used to forward messages from one node to another. Ethernet switches exhibit a substantial delay of typically several 10 µs for a packet having to pass through the switch, and this delay varies over time. To alleviate the deterioration caused by network switches due to the undefined amount of time a packet stays on the switch before it is forwarded, a mechanism that measures this variable duration on-the-fly is necessary for the SynUTC technology. This measurement is performed by another ASIC (Application Specific Integrated Circuit), which enhances standard Ethernet Switch functionality with a delay measurement mechanism to achieve highest precision and accuracy also in switched networks.

Furthermore the presented architecture allows the realization of IEEE 1588 boundary clocks, which interconnect different synchronization segments. The necessary PTP protocol stack runs on the 32-bit NIOS CPU, which is integrated along with the timestamping circuitry, an adder-based clock, and the IEEE 802.3 media access controllers.

7. Conclusion

An overview of a prototype system for high accuracy clock synchronization including the SoC chips and the embedded synchronization algorithms has been presented in this work.

The two discussed synchronization mechanisms, IEEE 1588 and SynUTC both are based upon taking highly accurate time information about leaving and incoming Ethernet packets into account.

However, with respect to fault tolerance the master-slave approach used in IEEE 1588 is disadvantageous.

Furthermore the accuracy intervals of the SynUTC technology represent additional information very useful for applications using industrial Ethernet.

And finally on-the-fly timestamp insertion into synchronization packets would avoid unnecessary network traffic in comparison to IEEE 1588 follow-up messages.

These issues could be considered in a later version or in a supplement of the IEEE 1588 standard.

References

[1] U. Schmid, *“Orthogonal Accuracy Clock Synchronization”, Chicago Journal of Theoretical Computer Science 2000(3)*, 2000, pp. 3-77. IEEE Computer, 1990, Vol. 23 (10), pp. 33-42.

[2] U. Schmid, M. Horauer, N. Kerö, *“How to Distribute GPS-Time over COTS-based LANs”, 31st Annual Precise Time and Time Interval (PTTI) Systems and Applications Meeting*, Dana Point - California, December 7-9 1999.

[3] Ulrich Schmid, Herbert Nachtnebel, *“Experimental Evaluation of High-Accuracy Time Distribution in a COTS-based Ethernet LAN”, Proceedings of the 24th IFAC/IFIP Workshop on Real-Time Programming (WRTP'99)*, Schloß Dagstuhl, Germany, May/June, 1999, pp. 59-69.

[4] Ulrich Schmid, Klaus Schossmaier, *“Interval-based Clock Synchronization”, Journal of Real-Time Systems 12(2)*, March 1997, pp. 173-228.

[5] M. Horauer, R. Höller, *“Integration of highly accurate Clock Synchronization into Ethernet-based Distributed Systems”, SSGRR 2002w*, 2002, ISBN 88-85280-62-5.

[6] R. Hoeller, M. Horauer, G. Griedling, N. Keroe, U. Schmid, K. Schossmaier, *“SynUTC - High Precision Time Synchronization over Ethernet Networks”, Proceedings of the 8th Workshop on Electronics for LHC Experiments*, 9-13 September 2002, Colmar, France, pages 428-432, ISSN 0007-8328, ISBN 92-9083-202-9.

[7] M. Horauer, K. Schossmaier, U. Schmid, R. Höller, N. Kerö, *“PSynUTC- Evaluation of a High Precision Time Synchronization Prototype System for Ethernet LANs”, Precise Time and Time Interval (PTTI)*, December 2002, Washington, USA.

[8] *IEEE Standard 802.3*, 2000 Edition, Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications, March 2000, New York, USA.

[9] *IEEE Standard 1588-2002*, Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, November 2002, New York, USA.

[10] D. Dietrich, T. Sauter, *“Evolution potentials for fieldbus systems”, IEEE International Workshop on Factory Communication Systems*, pp 343-350, Porto, Portugal, Sep. 6-8, 2000.

[11] S. Rüping, E. Vonnahme, J. Jasperneite, *“Analysis of Switched Ethernet Networks with different Topologies used in Automation Systems”*, D. Dietrich, P. Neumann, and H. Schweinzer, editors, Proceedings of the Fieldbus Technology Conference (FeT 99), pages 351 - 358, Magdeburg, 23 - 24 September 1999, Springer-Verlag.

[12] P. Otanez, J. Parrott, J. Moyne, D. Tilbury, “*The Implications of Ethernet as a Control Network”, Global Powertrain Conference*, Ann Arbor, MI, September 2002.

[13] J. Löser, H. Härtig, *“Real Time on Ethernet using off-the-shelf Hardware”, Proceedings of the 1st International Workshop on Real-Time LANs in the Internet Age (RTLIA 2002)*, Vienna, Austria, June 2002.

[14] N. Krommenacker, J. Georges, E. Rondeau, T. Divoux, Designing, *“Modeling and Evaluating Switched Ethernet Networks in Factory Communication Systems”, Proceedings of the 1st International Workshop on Real-Time LANs in the Internet Age (RTLIA 2002)*, Vienna, Austria, June 2002.