XIX IMEKO World Congress
Fundamental and Applied Metrology
September 6–11, 2009, Lisbon, Portugal

EVALUATION OF IEEE1588 APPLIED TO SYNCHRONIZED ACQUISITION IN MARINE SENSOR NETWORKS (MSN)

Joaquín del Rio[1], Daniel Toma[1], Antoni Mànuel[1], Helena Ramos[2]

[1] Sarti Research Group, Technical University of Catalonia in Vilanova i la Geltru, Barcelona, Spain
Joaquin.del.rio@upc.edu
[2] Instituto de Telecomunicaçoes, DEEC, Instituto Superior Tecnico, Lisboa, Portugal, hgramos@lx.it.pt

Abstract – Marine Sensor Networks (MSN) and ocean observatories are complex data acquisition systems where in many cases, synchronized acquisition is needed between the different networked nodes. In the development of OBSEA project, (EMSO CAC-2007-09) [1] distributed acquisitions nodes based in low power embedded systems have to be synchronized. This paper presents the evaluation results obtained using IEEE 1588 [2][3] standard as the synchronization protocol for the networked acquisition devices and a IM3220 microprocessor based embedded system.

Keywords: IEEE1588, MSN, Embedded System, Synchronization

1. INTRODUCTION

SARTI research group of Technical University of Catalonia is developing the OBSEA[1] project. OBSEA is an expandable offshore ocean cabled observatory to be installed in 2009, where embedded acquisition systems are connected to an Ethernet network. Applications where synchronised trigger at different nodes is needed will be implemented using embedded systems implementing IEEE 1588 as the synchronization protocol. This paper describes the implementation and presents measurements of the time-delays between Pulse per second (PPS) signals produced at different acquisition nodes by different Commercial of-the-shelf (COTS) Ethernet Switches in order to evaluate the performance of the synchronization protocol. Later on these PPS signals will be used to trigger the acquisition process in the networked nodes.

2. IEEE 1588 BASICS INTRODUCTION

IEEE 1588 defines the Precision Time Protocol (PTP) to synchronize "clocks" over the network.
Fig. 1 schematically describes its operation hierarchy in a master/slave communication over the network architecture and the Open System Interconnection (OSI) Reference Model. The PTP realises the Application Layers of the OSI reference model using User Datagram Packets (UDP) packets over Internet Protocol (IP) on the Ethernet network. Ethernet specifies the Physical Layer (PHY) and the Media Access Control (MAC) of the Data Link Layer of the OSI model.
The basic operations of PTP protocol are: frequency transfer, time transfer, basic operation of clock adjustment, influence of the network and the protocol stack.

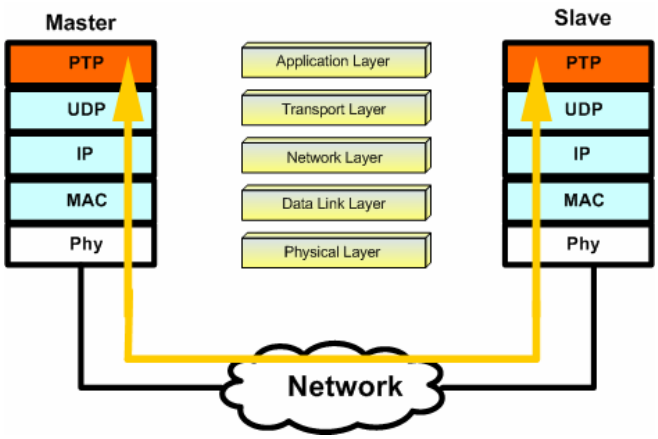

Fig. 1. PTP Layers.

The scope of frequency transfer operation is phase change rate (drift) compensation, realized by timestamped messages, called "Sync" messages, sent by the master to the slave consecutively to adjust the frequency (accelerate or delay the oscillator). The compensation is repeated regularly depending on the desired accuracy and on the oscillator stability.
The offset time between master and slave is calculated and corrected based on the round-trip delay measurement carried out by timestamped messages. First the master transmits to the slave a "Sync" message; then the salve sends a "DelayRequest" to the master, which is returned, and the round-trip delay is calculated using the timestamps).
In a network synchronized using the IEEE 1588 protocol each node must determine a priori which clocking source it should use. This operation is carried out by the Best Master Clock (BMC) algorithm which runs independently on all nodes and selects if the node is to be a time master that distributes its time to time slaves or not.

ISBN 978-963-88410-0-1

3. EXPERIMENTAL SETUP

In order to evaluate the performance when Precision Time Protocol is adopted for network synchronization the time delay between two PPS signals, F1 and F2, is measured.

3.1. Hardware Description

One PPS signal (F1) is generated on a IM3220 microprocessor from IMSYS [5]. The IM3220 is a dedicated controller for networked applications that combines the features of traditional CISC architectures and efficient use of resources with FPGA flexibility for a pre-defined application – time synchronization. It comes with IEEE 1588 support for a Precise Time Engine and also with an API programmable IEEE 1588 software stack.

A computer running WindowsXP and a National Instruments PCI card PCI1588 generates the other PPS signal (F2) and works as a Master Clock in the system. The National Instruments PCI-1588 uses the IEEE 1588-2002 precision time protocol to synchronize clocks and events with other IEEE 1588-based instruments, computers or embedded systems. The NI PCI-1588 uses a standard RJ-45 Ethernet plug and CAT 5 cabling. It is capable of operating as either an IEEE 1588 master or slave clock module. An onboard FPGA automatically adjusts the frequency and phase of an onboard IEEE 1588 clock [6].

In order to measure PPS signals delays, PPS signals F1 and F2 are combined using a XOR gate that generates submicroseconds pulses when its input signals are not synchronous. Figure 2 shows the experimental setup used. A COTS Ethernet Switch is used to implement the Ethernet network. Delay measurements are made with two different models: an industrial switch model RuggedCOM from Industrial Strengh Networks and a Base Line Switch 2016 from 3COM. The Agilent 53132A Universal Frequency Counter, with 12 digits per second and 150ps time interval resolution, is used to measure the pulse width generated by the XOR gate and a computer with a LabVIEW application stores the measurements in a data file.

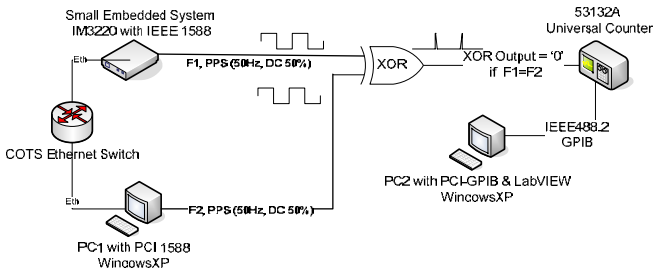

Fig. 2. Setup for F1 to F2 delay measurements.

Fig. 3 shows a oscilloscope capture of the XOR gate output for a delay between the two PPS signals, F1 and F2 of about 30ns.

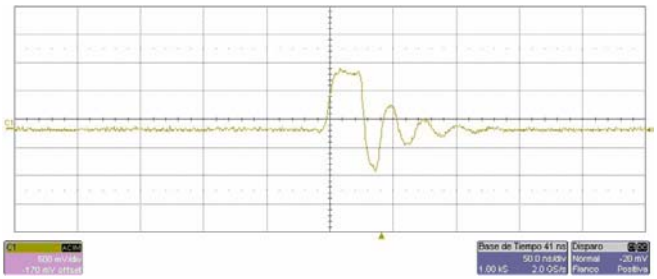

Fig 3. XOR gate output for PPS signals delay of 30ns.

3.2. Software configuration

In order to program the IM3220 based embedded system a high level programming language (C) is used. Imsys Developer platform is used as a compiler and debugging application.

Figure 4 shows the PTP() function developed to activate the PTP IEEE1588 protocol and the F1 PPS signal generation using the API control software functions for IEEE1588.

```
#include <string.h>
#include "ptp.h"
#include "time.h"
#include "rubus.h"
    int control_start=1;       //PTP start Flag
     void ptp(void)
     {
   struct PTP_Time_T startTime;
   struct PTP_Time_T width;
   struct PTP_Time_T period;
   startTime.seconds_ = 0;    //Define the PPS start time
   startTime.nanoseconds_ =0;
   width.seconds_ = 0;    //Define the width of PPS pulse
   width.nanoseconds_ = 10000000;
   period.seconds_ = 0; //Define the PPS pulse period
   period.nanoseconds_ =20000000;

    unsigned long pulsecount= 0;
    PTP_Start(control_start);    //Start the PTP Engine
     //Get PTP time
    PTP_ClockItf_T_getTime((PTP_Time_T*) &startTime);
     //delay to start the PPS
    startTime.seconds_ = startTime.seconds_ + 10; signal
     // F1 PPS signal  Generation
    PTP_PhysIfs_T_PulseTrain((PTP_Time_T*)&startTime,(PTP_Time_T*)
    &width,(PTP_Time_T*)&period,pulsecount);
     }
```

Fig 4. PTP engine activation and PPS signal generation in Imsys Embedded System as an example of the code developed.

After configuring the Master Clock and activating the PTP synchronization protocol the F2 PPS signal was generated using the PCI1588 card. Fig. 5 shows de LabVIEW code used to configure and generate a PPS signal with a frequency of 50Hz and a 50% of duty cycle. Table 1 flowcharts the program.

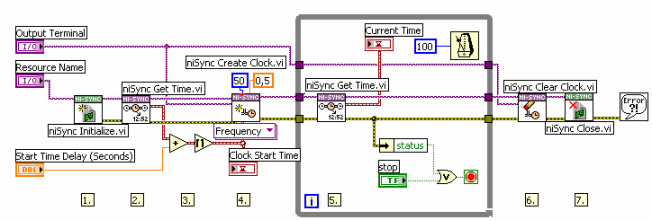

Fig 5. LabVIEW Code programmed to configure and generate F2 PPS Signal from PCI1588 National Instruments card.

Table 1. Flowchart of the program to configure and generate F2.

step	Description
1	Create a new NI-SYNC driver session
2	Read de current 1588 time
3	Define when the clock signal will be generated by adding a delay to the current 1588 time and rounding up to the nearest seconds boundary
4	Program the frequency of F2 to 50Hz and the start time of the clock to be generated
5	Display the current 1588 time
6	Clear the clock wich stops the F2 PPS clock generation and release the resource
7	Close the driver session

F1 and F2 have the same parameters. The PTP synchronization engine had to be started in both sources before recording the measurement delays. Code to do that is not included in this paper.

4. EXPERIMENTAL RESULTS

4.1 Drift without PTP

In order to justify the need of a time synchronization protocol, the time delay between the two PPS signals (F1 and F2) with a frequency of 50 Hz and duty cycle 50%, generated respectively by the Imsys controller and by the PCI 1588 NI card, is measured after combining the signals at a XOR gate. The results obtained for the drift are presented in Fig. 6.

Fig. 6 shows how the delay increases in function of time without the application of a synchronization protocol. Each platform is generating a 50Hz clock signal independently but, due to drifts in their main clocks, signals are drifting from one other. As depicted, clocks drift 10ms each 1100 seconds. This represents a drift about 9µs/s. As Fig. 6 shows, this drift without synchronization is apparently constant.

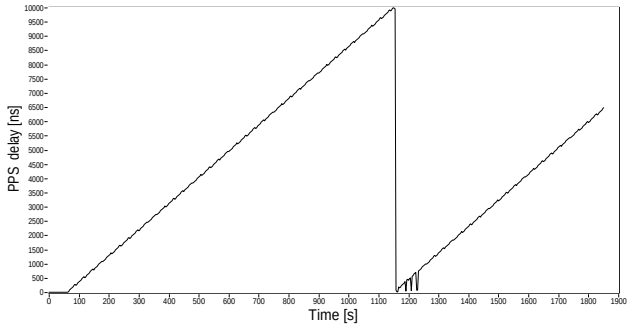

Fig 6. Drift between PPS signals without PTP synchronization protocol

4.2 Delay measurements with PTP

Precision Time Protocol IEEE1588 is used to synchronize the different clock sources. Delays between F1 and F2 had been measured with two different COTS switches. Measurements obtained using *model a* are presented in Figs. 7 and 8.

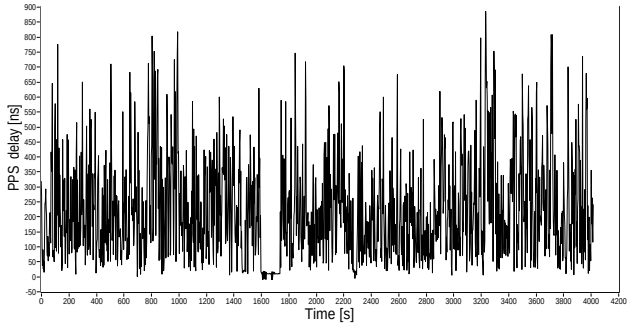

Fig 7. PPS signals delay evolution with PTP IEEE1588 using COTS Ethernet Switch *model a*.

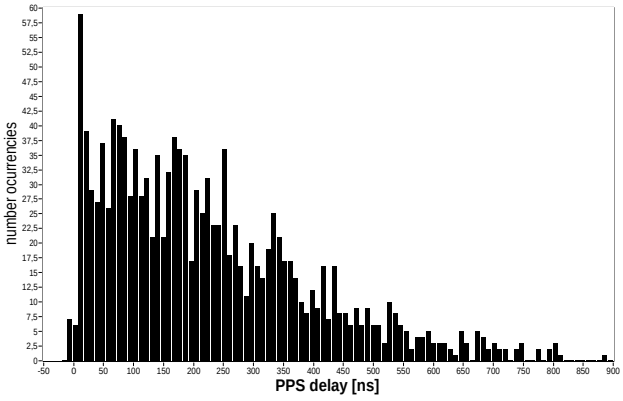

Fig. 8. PPS signals delay histogram with PTP IEEE1588 using COTS Ethernet Switch *model a.*

Delays with Ethernet switch *model a*, as it is shown in Figs. 7 and 8 have a mean value of about 224ns with a standard deviation of 171ns and a maximum delay of 884ns. The time delay between signals is always lower than 1µs.
Figs. 9 and 10, show similar measurements using another COTS Ethernet Switch designated as *model b*.

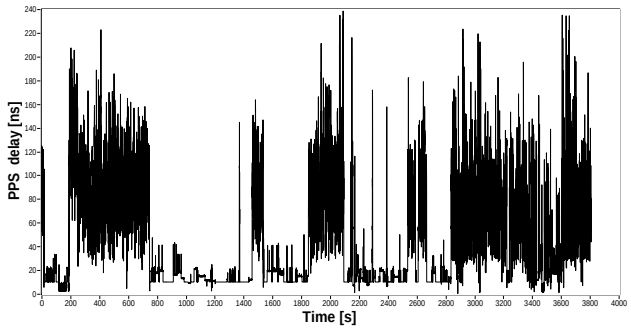

Fig. 9. PPS signals delay evolution with PTP IEEE1588 using COTS Ethernet Switch *model b*.

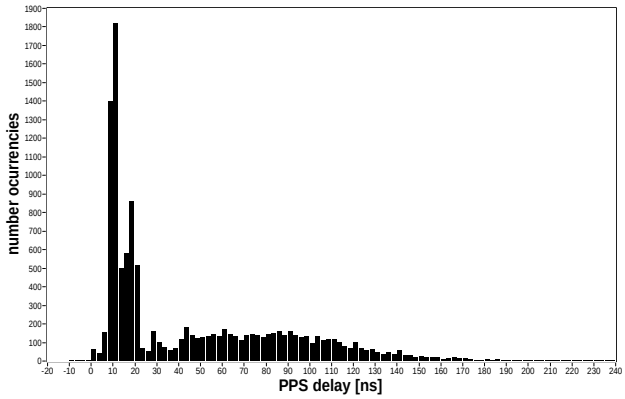

Fig. 10. PPS signals delay histogram with PTP IEEE1588 using COTS Ethernet Switch *model b*.

The results presented when COTS Ethernet Switch *model b* is being used show a time delay mean value lower than the corresponding values measured with Switch *model a*. The time delay mean value is 46ns with a standard deviation of 42ns and a maximum value of 238ns.

It is clear when a comparison between Fig.6 and Figs. 7, 8, 9 and 10 is made that when PTP engine is running in both platforms, a significant improvement in the time delays between the two PPS signals F1 and F2 is achieved.

5. CONCLUSIONS

An experimental setup and the results obtained have been presented in order to evaluate delays between trigger clock signals in two acquisition nodes in an Ethernet cabled sensor network. IEEE1588 is used to synchronize trigger clock signals used to trigger front end of the analogue to digital converters. Delay measurements are made building an Ethernet network with two different switches. Table 2 shows a resume of the measurement in function of switch model.

Table 2. Experimental results obtained for two PPS signals delay.

	Switch a	Switch b
Mean Delay [ns]	224	46
Standard Deviation [ns]	171	42
Maximum Delay [ns]	884	238

With these results is important to remark that IEEE1588 standard is a valid strategy to synchronize distributed nodes in a Marine Sensor Network where Ethernet is used and synchronization between different nodes as a Geophones or Hydrophones acquisition systems.

Measurements about how temperature variations affect the time synchronization are being and will be presented at final paper.

REFERENCES

[1] C. Artero, M. Nogueras, S. Shariat-Panahi, A. Mànuel, P. Santamaría, J. Cadena, O. Gualdo, "Diseño del sistema de control y adquisición de datos del Observatorio Submarino ExpAndible (OBSEA)", *SAAEI08*, Cartagena, Spain, Sept. 2008.

[2] Introduction to IEEE 1588. [Online]. Available at: http://ieee1588.nist.gov/intro.htm.

[3] Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, IEEE STD 1588-2008, IEEE Instrumentation and Measurement Society, TC-9, The Institute of Electrical and Electronics Engineers, Inc., New York, NY, 24 July 2008.

[4] Person R, "Long term multidisciplinary deep sea observatories: how to find compromise between scientific needs, technological capabilities and financial resources: the ESONET contribution", *Proceedings of UT2007*, pp. 292-298, Tokyo, Japan, April 2007.

[5] IMSYS 3220 microprocessor information: http://www.imsys.se/products/im3220.htm available on January 14th 2009.

[6] National Instruments PCI1588 Precision Time Protocol Interface: http://sine.ni.com/nips/cds/view/p/lang/en/nid/202345 available on January 14th 2009.